AD-765 331

ASP: A STATISTICAL PACKAGE IN APL

Francis J. Anscombe

Yale University

Prepared for:

Office of Naval Research

July 1973

DISTRIBUTED BY:

NTIS
National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

AD 765331

YALE UNIVERSITY

Department of Statistics

Technical Report No. 29

July, 1973

Francis J. Anscombe

ASP: A Statistical Package in APL

[Draft appendix to the forthcoming book, "Statistical Computing with APL"]

This research was supported by the Army, Navy, Air Force and NASA under a contract administered by the Office of Naval Research. Reproduction in whole or in part is permitted for any purpose of the United States Government.

Contract number: N00014-67-A-0097-0014

Task number: NR 042-242

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Yale University, Department of Statistics	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

"ASP": A STATISTICAL PACKAGE IN APL

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Francis J. Anscombe

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
July, 1973	27	

8a. CONTRACT OR GRANT NO. N00014-67-A-0097-0014	9a. ORIGINATOR'S REPORT NUMBER(S) Technical Report No. 29
b. PROJECT NO.	
c. Task No. NR 042-242	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES Reproduced from best available copy.	12. SPONSORING MILITARY ACTIVITY Office of Naval Research Arlington, VA

13. ABSTRACT

The report presents a collection of computer programs for statistical analysis of data, written in the programming language APL, for execution in IBM's interactive system APL\360. A short introduction comments on special features of computing for statistical analysis and suggests some precautions in the writing of such programs. This material is intended to appear eventually as an appendix to the author's forthcoming book, "Statistical computing with APL".

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

Unclassified
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Statistical computing						
APL						

ASP: A Statistical Package in APL

[Draft appendix to the forthcoming book, "Statistical Computing with APL"]

Designing a statistical package -- a set of programs for statistical analysis of data -- poses problems, some of them common to all computing, some perhaps peculiar to statistics.

Features of computing for statistical analysis, not common to all computing, are that many diverse operations are carried out on different sets of data, that several basic operations may be applied to any particular set of data, and that only after a basic operation has been applied and the output examined can the user see (usually) whether the operation was appropriate and satisfactory. Statistical analysis is a process of trial and error, not to be specified simply and completely in advance. A statistical package is therefore naturally thought of as a kit of tools rather than as a single machine. Ingenuity may be devoted to organizing the tools into one or a few machines, but if, as with APL\360, the computing is conversational, flexibility is to be preferred to grandeur, unless indeed the user needs to be shielded from programming.

A prewritten package of programs is attractive to a user who does not consider himself expert in the language in which they are written. Programs in APL\360, such as those presented here, demand some knowledge of the language and especially some knowledge of the system commands and of how to manage workspaces. Given modest knowledge of that kind, the programs can be successfully used by beginners in computing who would not readily have composed equivalent programs themselves. Ultimately, however, the only good reason for using APL for statistical work is to have complete freedom to do what one wishes. These programs should be thought of as suggestions for how to proceed and as a source of examples of programming technique, rather than as prescriptions for good statistical practice. The programs are there to be altered and adapted. They are not locked.

Programs can have at least four technical virtues, apart from their content and purpose, namely (i) clarity to the reader, (ii) speed in execution (CPU time and connect time), (iii) economy in space required for execution, (iv) economy in space required for storage. All four are important, though their relative importance is arguable. Any actual program is at best a compromise -- one virtue may be traded for another. Only if a program is to be used often or kept a long time are the economies worth much thought, once a few rudimentary principles have been learned: loops should usually be avoided if possible, numerical data should not be stored permanently in floating-point form, and things like that.

Programs (functions) and data sets to be kept some time or seen by others should be documented. The very least is to insert a line or two of comment in the definition of a function (as for example in RNORMAL shown below in display-sheet 3 -- anyone who knows what is meant by random normal deviates should have no difficulty in using this function after seeing a print-out of the definition). My usual practice is to provide minimal information about a data set under a name (of a variable or function) stored in the same work-space, consisting of the letters WHAT joined to the name of the data set, and information about a function under a name beginning with the letters HOW . Such documentation occupies least storage space, usually, if it is a variable consisting of a character vector. Examples are displayed below. My practice is to conclude a HOW-variable with a date, the date of the most recent perceptible change made in it. The user may quickly discover the date of a HOW-variable by calling for the last 13 characters, and only display the whole if the date is later than that of his last print-out.

Such documentation does not attempt to explain statistical principles. More extended information, explanations of purpose and illustrations of use, can be issued as duplicated reading matter, not kept in the APL system.

Some precautions

As far as conveniently possible, functions should be written so that they will not break down (be suspended) in execution, but rather will terminate with a warning message if something is wrong. At least to begin the definition of a function by testing the arguments (if there are any) for compatibility seems to be good practice. This not only takes care of a common cause of failure in execution, but greatly assists a reader who is not fully informed about what the function does. If the user is puzzled to get the message 'NO GO.' when he tries to execute the function, he can examine the first two or three lines of the definition to see what requirements have been placed on the arguments.

Consider for example the function FIT shown below (display-sheet 3). As HOWFIT explains, this function is intended to subtract from a given vector (Y, say) its linear regression on another given vector (X, say). It could be used to detrend a time series by removing its linear regression on the time variable. The means, the regression coefficient and its conventionally estimated standard error, are printed out for information. The number of readings N is defined in line [1], and thereafter the desired calculations and output are specified in lines [3], [4], [5], [8], [9], [10], in a manner that should not prove difficult to read. Line [1] is primarily concerned with testing that the arguments are indeed vectors of equal length not less than 3, and if they are not line [2] causes a message to be displayed and execution to stop. Line [6] tests whether all members of the first argument (the "independent" variable) are equal. If they are, the vector X is this line consists of equal numbers almost or exactly zero, and the regression calculation in line [8] should not (perhaps cannot) be executed, and so execution of the function stops at line [7]. If execution stops at line [2], the explicit result Z is empty. If execution stops at line [7], the explicit result is just the second argument minus its mean. Only if both these traps are passed does the explicit result become the residuals from the fitted regression, as intended.

This function FIT has not been protected from every conceivable cause of breakdown. If the arguments were vectors of equal length not less than 3, but if either or both were vectors of characters rather than of numbers, execution would be suspended at line [3] or [4] where the sum reduction is called for. There ought to be a primitive monadic function in APL that would yield the type of its argument, possibly (for example) 0 for characters, 1 for single bits, 2 for 4-byte integers, 3 for 8-byte floating-point numbers. As things are now, one may test whether an array X is numerical with the expression

```
0=1↑0↑,X .
```

This works because APL\360 distinguishes empty numerical vectors from empty character vectors, but that has the appearance of being an accident of the implementation, not obviously required by any general syntactic principle; in other contexts APL\360 ignores the type of empty arrays. To supply a character argument to a function where a numerical argument is intended, or vice versa, is no doubt an unusual error in execution, scarcely worth guarding against. At any rate, it has not been guarded against here.

Some of the functions in this collection, such as DOWNPLOT , require that one or more global variables should have been defined before the function is executed. There seems to be no way at present to test in the function definition whether that is so, without a suspension if in fact the required global variables have not been defined.

Another possible reason for breakdown in execution of a correctly written function is that there is insufficient room in the active workspace. If the space requirement were determined, as a function of size of arguments and any other relevant variables, this also could be tested for. I have generally not done so, but an example of a possibility of this sort may be seen in the function FILTER (display-sheet 9). The meat of the function is in the single line [9], which executes fast but needs room, sometimes a lot of room. The last three lines in the definition constitute an alternative program to line [9], slower to execute because of the loop but needing less space. In line [8] a test is made of available space, with a branch to [10] if [9] cannot be executed. (For this calculation it is supposed that the argument X consists of floating-point numbers, which it generally does in practice.) Breakdown in execution because of lack of space has not been guarded against completely, there may be too little room in the active workspace even for the more economical program, but the worst space problem in FILTER has been taken care of.

Suspensions of function execution will occur. Sometimes the user's attention is so fixed on remedying the cause or taking other action prompted by the suspension that he omits to terminate execution of the suspended function. Then the function's local variables supersede any global objects with the same names. The most annoying effect of this kind comes from statement labels, which cannot (during suspended execution of a function) be treated like ordinary names of variables and assigned new values. Preferably statement labels should be unlike any other names used. Accordingly all statement labels here begin with an underscored L . If the user refrains from ever assigning to a variable or function a name beginning with underscored L (no very irksome constraint, since underscoring is troublesome anyway), there can be no confusion.

Possibly there should be some similar distinguishing of all local variables in defined functions. That, however, is not so easy to achieve, partly because in developing a function the author should feel free to change the status of variables between local and global. For economy's sake, no local variable should have a name longer than three characters*. If global variables and functions intended for permanent storage are given descriptive names more than three characters long (as is my usual but not invariable practice), at least there will be no confusion between them and local variables.

A particular kind of suspension occurs with quad input, or as the User's Manual calls it, evaluated input. Only three of the functions shown here have this feature: INTEGRATE, SCATTERPLOT, TSCP. In answering the first question in SCATTERPLOT or TSCP , the user will normally type two numbers, as suggested; but if instead he gives the name of a previously defined 2-element vector, this must be different from the names of all eleven local variables (not counting statement labels) in the function. The same applies to the third question in TSCP , where now there may be greater temptation to reply with the name of a previously defined character vector (Z would be permissible, or any name with more than one character). Similarly, if the function asked for in INTEGRATE is previously defined, its name must not be any one of A, H, N, Z (it could happily be F, or any name with more than one character).

----

*There is a small unobvious economy in storage of functions to be derived from keeping the names of local variables short, mostly single characters. That is that in a collection of functions the same names of variables will occur again and again, for there are only 26 letters in the alphabet to choose from. In storage each different name appears just once in the symbol table, which therefore needs to be large enough to accommodate all the different names in the workspace -- the names of groups, functions and variables, including the variables and statement labels in the function definitions. In the functions shown here, a handful of one-letter names account for many of the local variables; each appears several times in the definitions, but only once in the symbol table. Therefore these functions can be stored in workspaces with symbol tables substantially reduced below the standard setting.

Miscellaneous remarks

Perhaps a word should be said about some of the algorithms, with reference to their speed in execution. One should beware of supposing that comparisons of alternative algorithms, executed in compiled Fortran, are relevant to apparently similar algorithms executed in APL\360. The richer vocabulary of primitive functions in APL, and the different handling of arrays, can lead to different conclusions. Because of its brevity, the function RNORMAL seems to be about the fastest possible generator of random normal deviates in APL. The function HA is a slow Fourier transform, but performs well with series of modest length, such as can be accommodated in the customary 32K byte workspace; and it leaves the user completely indifferent to the prime-power factorization of the length of the series. (A fast Fourier transform will no doubt eventually be added to the collection.) The loop at the end of SCATTERPLOT could be avoided by an outer product, but that seems slow in execution and certainly requires considerable space -- a counter-example to the general principle of avoiding loops if possible.

Most of these functions have been developed independently of anyone else's programming. An exception is INTEGRATE, based on a program for integration by Simpson's Rule due to John Richardson (APL Quote Quad, 2/3 (1970), 26-27). The function JACOBI was developed from two APL programs written by others, but is not now very close to either source. K. W. Smillie's collection of statistical programs, STATPACK2, provided many interesting examples of APL technique, but I have deliberately refrained from actually using his programs in order not to be influenced in statistical thinking.

The display-sheets following show function definitions and documentary variables in the workspaces ASP and ASP2. In alphabetical order, with the sheet number on which they appear, they are:

Documentary variables

Function definitions

```
      )LOAD ASP
SAVED  21.48.43 07/17/73
      )FNS
ANALYZE DOWNPLOT           EFFECT  FIT      INIF     INTEGRATE        JACOBI  NIF
REGR    REGRINIT           RNORMAL ROWCOL   ROWCOLDISPLAY   ROWCOLPERMUTE    ROWPLOT
SAMPLE  SCATTERPLOT        SHOW    STDIZE   STRES    SUMMARIZE        VARIANCE
      )VARS
DESCRIBE        HOWFIT  HOWINTEGRATE    HOWJACOBI       HOWPLOT HOWREGRESSION
HOWSUMMARIZE    A       B       C
      )GRPS
NWAYGROUP       REGRESSIONGROUP ROWCOLGROUP
      )GRP NWAYGROUP
ANALYZE EFFECT  C
      )GRP REGRESSIONGROUP
REGR    REGRINIT        SHOW    STRES   VARIANCE        C
      )GRP ROWCOLGROUP
ROWCOL  ROWCOLDISPLAY   ROWCOLPERMUTE   C
```

```
      DESCRIBE
```

ASP - A STATISTICAL PACKAGE IN A PROGRAMMING LANGUAGE

↑↑↑ . ↑ ↑.::..::..: ↑.::..: .. * *::..::..:: *::..::!

SOME PROGRAMS FOR STATISTICAL ANALYSIS BY F. J. ANSCOMBE, WHO WILL APPRECIATE COMMENTS AND ERROR REPORTS. (PHONE: 203-432-4752.)

'HOWPLOT' AND 'HOWTRIPLEPLOT' DESCRIBE PLOTTING FUNCTIONS.

SEE 'HOWSUMMARIZE' FOR SIMPLE SUMMARY STATISTICS, 'HOWFIT' AND 'HOWREGRESSION' FOR LEAST-SQUARES REGRESSION, AND 'HOWHUBER' FOR ROBUST REGRESSION. SEE ALSO THE FUNCTIONS 'ROWCOL' AND 'ANALYZE'. 'HOWFILTER' AND 'HOWHARMONIC' DESCRIBE FUNCTIONS FOR ANALYSIS OF TIME SERIES; SEE ALSO 'AUTOCOV'. 'HOWCONTINGENCY' DESCRIBES FUNCTIONS FOR ANALYSIS OF CONTINGENCY TABLES.

'NIF' AND 'INIF' ARE THE NORMAL INTEGRAL FUNCTION AND ITS INVERSE. VARIOUS MATHEMATICAL OPERATIONS ARE DESCRIBED IN 'HOWJACOBI', 'HOWINTEGRATE', 'HOWMAX'.

THE FUNCTIONS 'ROWCOL', 'ANALYZE', 'EFFECT', 'AUTOCOV', 'SAMPLE', 'RNORMAL', 'NORMALTEST' ARE DESCRIBED IN CHAPTER 7 OF <u>STATISTICAL COMPUTING WITH APL</u>.

THESE FUNCTIONS RESIDE IN WORKSPACES NAMED 'ASP' AND 'ASP2'. SOME OF THE FUNCTIONS ARE ORGANIZED IN GROUPS. BEGIN BY LISTING ALL FUNCTIONS, VARIABLES AND GROUPS IN EACH WORKSPACE.

1-ORIGIN INDEXING IS ALWAYS SUPPOSED. SOME FUNCTIONS NEED THE GLOBAL SCALAR CHARACTER VARIABLE C CAUSING A CARRIER RETURN. C IS INCLUDED IN EVERY GROUP. C SHOULD BE SPECIFIED OR COPIED WHEN ANY OF THESE FUNCTIONS (BUT NOT A GROUP) IS COPIED. THE SCALAR VARIABLE B CAUSES A BACKSPACE. THE VECTOR A IS THE ALPHABET, FIRST PLAIN THEN UNDERSCORED.

DEPT. OF STATISTICS, YALE UNIV., NEW HAVEN, CT. 06520 - 11 JULY 1973 ::..::..::!

HOWSUMMARIZE

SUMMARY STATISTICS OF A DATA SET
SUMMARIZE Z

THE ARGUMENT (Z) MAY BE ANY NUMERICAL ARRAY HAVING AT LEAST 4 MEMBERS. MEASURES OF LOCATION, SCALE AND SHAPE OF DISTRIBUTION ARE PRINTED OUT. SHAPE IS INDICATED BY FREQUENCIES OF THE DATA IN THE SIX INTERVALS INTO WHICH THE REAL LINE IS DIVIDED BY THE MEAN + ¯2 ¯1 0 1 2 × *THE STANDARD DEVIATION OF THE DATA. EXPECTED COUNTS IN THESE INTERVALS FOR A RANDOM SAMPLE OF THE SAME SIZE TAKEN FROM A NORMAL DISTRIBUTION WITH THE SAME MEAN AND VARIANCE ARE GIVEN FOR COMPARISON. SHAPE IS ALSO INDICATED BY FISHER'S SKEWNESS AND KURTOSIS MEASURES, WITH THEIR STANDARD ERRORS FOR A RANDOM SAMPLE OF THE SAME SIZE FROM A NORMAL DISTRIBUTION.*

28 *JUNE* 1973

```
    ∇ SUMMARIZE Z;A;D;K;M;N;S;W
[1]   →2+4≤N←ρZ←,Z
[2]   →ρ⎕←'THE ARGUMENT MUST BE AN ARRAY WITH AT LEAST 4 MEMBERS.'
[3]   C,'NUMBER OF READINGS:  ';N
[4]   'EXTREMES:  ';(Z←Z[⍋Z])[1,N]
[5]   'QUARTILES:  ';⌽A←0.5×(+/Z[N+1-A]),+/Z[A←(⌊1+N÷4),⌈N÷4]
[6]   'MEDIAN:  ';0.5×+/Z[(⌊1+N÷2),⌈N÷2]
[7]   'MEAN:  ';M←(+/Z)÷N
[8]   C,'INTERQUARTILE RANGE:  ';-/A
[9]   'VARIANCE:  ';K←(S←+/W←Z×Z←Z-M)÷N-1
[10]  'STANDARD DEVIATION:  ';D←K*÷2
[11]  A←0.1×⌊0.5+0.0001×N× 2275 13591 34134
[12]  C,'FREQUENCIES:  ';+/(¯4+⍳6)∘.=2⌊¯3⌈⌊Z÷D;',  WITH FITTED NORMAL EXPECTATIO
      NS  ';A,⌽A
[13]  'SKEWNESS MEASURE G1:  ';(÷K×D)×÷/(+/W×Z),N+ ¯1 0 ¯2 ;',  WITH NORMAL S.E.
        ';(÷/6,N+ ¯2 0 1 ¯1 3)*÷2
[14]  A←(÷K×K)×(÷/(+/W×W),N+ ¯1 0 ¯2 1 ¯3)-3×S×S÷(N-2)×N-3
[15]  'KURTOSIS MEASURE G2:  ',A;',  WITH NORMAL S.E.  ';(÷/24,N+ ¯3 0
      ¯2 ¯1 3 ¯1 5)*÷2;C
    ∇
```

HOWFIT

SIMPLE REGRESSION
Z←X FIT Y

THE TWO ARGUMENTS MUST BE VECTORS OF EQUAL LENGTH NOT LESS THAN 3.

THE MEANS OF THE TWO ARGUMENTS, AND THE REGRESSION COEFFICIENT OF THE SECOND ARGUMENT ON THE FIRST, WITH ITS CONVENTIONAL ESTIMATED STANDARD ERROR, ARE DISPLAYED. THE EXPLICIT RESULT IS THE VECTOR OF RESIDUALS.

30 *MAY* 1973

```
    ∇ Z←X FIT Y;B;N;S;U;V
[1]   →2+(N=+/ρY)∧(3≤N←+/ρX)∧(1=ρρY)∧1=ρρX
[2]   →ρ⎕←'ARGUMENTS MUST BE VECTORS OF EQUAL LENGTH NOT LESS THAN 3.',Z←''
[3]   X←X-U←(+/X)÷N
[4]   Z←Y-V←(+/Y)÷N
[5]   'MEANS ARE  ';U,V
[6]   →8-∧/X=1↑X
[7]   →ρ⎕←'NO REGRESSION CALCULATED.'
[8]   'REGRESSION COEFFICIENT IS  ';B←(X+.×Z)÷S←X+.×X
[9]   S←(Z+.×Z←Z-B×X)÷S×N-2
[10]  '    WITH ESTIMATED STANDARD ERROR  ';S*0.5;'  (';N-2;' D.F.)'
    ∇
```

```
    ∇ Z←RNORMAL N;V
[1]   →2+∨/(N≥1)∧(N=⌊N)∧1=ρN←,N
[2]   →0,ρ⎕←'NO GO.',Z←''
[3]   Z←(÷2147483647)×?(2,⌈N÷2)ρ2147483647
[4]   Z←N↑(, 1 2 ∘.○Z[2;]×○2)×V,V←(¯2×⍟Z[1;])*0.5
[5]   ⍝  N RANDOM NORMAL DEVIATES BY BOX-MULLER METHOD.
    ∇
```

```
    ∇ Z←N SAMPLE CP
[1]   →2+(∧/,(N=⌊N)∧1≤N)∧(1≥ρρN)∧∧/(1≥CP)∧0≤CP←,CP
[2]   →ρ⎕←'NO GO.',Z←''
[3]   Z←+⌿CP∘.<(÷2147483647)×?Nρ2147483647
[4]   ⍝  SAMPLE OF SIZE N FROM A DISTRIBUTION OVER NONNEGATIVE INTEGERS HAVING C
      UMULATIVE PROBABILITIES CP.
    ∇
```

HOWREGRESSION

MULTIPLE REGRESSION BY STAGES AND EXAMINATION OF RESIDUALS

(1) X REGRINIT Y
(2) REGR L
(3) SHOW V
(4) STRES
(5) VARIANCE

'REGRINIT' IS USED ONCE AT THE OUTSET TO SET UP GLOBAL VARIABLES FOR 'REGR'. 'REGR' PERFORMS REGRESSION ON ONE OR MORE DESIGNATED INDEPENDENT VARIABLES, AND MAY BE CALLED REPEATEDLY. 'SHOW' MAY BE USED AT ANY STAGE TO OBTAIN SUMMARY INFORMATION ABOUT A VECTOR. 'STRES' GIVES STANDARDIZED RESIDUALS OF THE DEPENDENT VARIABLE(S) FOR USE IN SCATTERPLOTS. 'VARIANCE' YIELDS THE CONVENTIONAL ESTIMATED VARIANCE MATRIX OF THE REGRESSION COEFFICIENTS.

(1) REGRINIT. THE FIRST ARGUMENT (X) IS A MATRIX WHOSE COLUMNS LIST VALUES OF THE INDEPENDENT (EXOGENOUS, PREDICTOR) VARIABLES. USUALLY ONE COLUMN IS ALL 1'S, AND ITS INDEX NUMBER IS THE ARGUMENT IN THE FIRST CALL OF 'REGR'. EACH COLUMN OF X SHOULD HAVE BEEN MULTIPLIED BY A POWER OF 10 SO THAT THE UNIT PLACE IS THE LAST SIGNIFICANT ONE. WHEN A COLUMN OF X-RESIDUALS HAS SUM OF SQUARES LESS THAN (NU÷4) IT IS JUDGED TO BE TOO CLOSE TO ROUND-OFF ERROR TO BE WORTH USING.

THE SECOND ARGUMENT (Y) IS EITHER A VECTOR LISTING VALUES OF ONE DEPENDENT VARIABLE OR A MATRIX WHOSE ROWS LIST VALUES OF SEVERAL DEPENDENT VARIABLES TO BE STUDIED IN PARALLEL. THE NUMBER OF OBSERVATIONS OR DATA SETS IS EQUAL TO

```
1↑ρX ↔ ¯1↑ρY
```

GLOBAL VARIABLES IN 'REGRINIT' AND 'REGR':

- **P** THE NUMBER OF INDEPENDENT VARIABLES, (ρX)[2] .
- **IND** A LOGICAL VECTOR OF LENGTH P SHOWING WHICH INDEPENDENT VARIABLES HAVE BEEN BROUGHT INTO THE REGRESSION RELATION (INITIALLY ALL 0'S).
- **NU** THE NUMBER OF RESIDUAL DEGREES OF FREEDOM (INITIALLY THE NO. OF OBS.).
- **RX** TRANSFORMED OR RESIDUAL X-VARIABLES (INITIALLY EQUAL TO X).
- **RY** RESIDUALS OF Y (INITIALLY EQUAL TO Y).
- **B** REGRESSION COEFFICIENTS (VECTOR OR MATRIX) OF Y ON COLUMNS OF X CONSIDERED SO FAR. (B APPEARS ONLY IN THE OUTPUT OF 'REGR'.)
- **RB** DITTO WITH RX IN PLACE OF X (INITIALLY ALL 0'S).
- **TRX** P-BY-P MATRIX TRANSFORMING X TO RX. ⍉RX ↔ TRX+.×⍉X ; B ↔ RB+.×TRX . THE FITTED VALUES ARE Y-RY ↔ B+.×⍉X ↔ RB+.×⍉RX .
- **ISS** INITIAL SUM OF SQUARES (APPEARING ONLY IN THE OUTPUT OF 'REGRINIT').
- **RSS** RESIDUAL SUM OF SQUARES (APPEARING ONLY IN THE OUTPUT OF 'REGR').

AFTER 'REGRINIT' IS EXECUTED IT CAN BE ERASED, AS ALSO ITS ARGUMENTS (WHICH SHOULD BE STORED FOR COPYING LATER WHEN NEEDED).

(2) REGR. THE ARGUMENT (L) IS A SCALAR OR VECTOR LISTING THE INDEX NO.(S) OF THE INDEPENDENT VARIABLE(S) TO BE BROUGHT NEXT INTO THE REGRESSION.

RENAME RSS OR ANY OTHER GLOBAL OUTPUT VARIABLE TO SAVE IT, BEFORE AGAIN CALLING 'REGR'.

THE POSSIBLE REDUCTIONS IN RSS TABULATED IN THE PRINTED OUTPUT SHOW THE EFFECT OF BRINGING ONE FURTHER INDEPENDENT VARIABLE INTO THE REGRESSION, AND MAY BE USED TO GUIDE THE NEXT CALLING OF 'REGR'.

THE SO-CALLED MODIFIED GRAM-SCHMIDT ALGORITHM IS USED. SEE

ANSCOMBE: J. ROYAL STATIST. SOC. B 29 (1967), 1-52, ESPECIALLY SECTION 1.4.
BJÖRCK: BIT - NORD. TIDSKR. INFORMATIONSBEHAND. 7 (1967), 1-21.

IF YOU WANT TO FOIL THE (NU÷4) PROVISION MENTIONED ABOVE, REPLACE THE '4' IN LINE [1] BY A LARGER NUMBER.

(3) <u>SHOW</u>. THE ARGUMENT (V) IS A VECTOR, SUCH AS RY (IF A VECTOR) OR A ROW OF RY (IF RY IS A MATRIX) OR A COLUMN OF RX OR DIAGQ (SEE 'STRES').

NU IS USED AS DIVISOR IN FINDING THE R.M.S. VALUE. IF 'SHOW' IS USED OUTSIDE THIS REGRESSION CONTEXT, NU MUST BE SPECIFIED FIRST.

THE FREQUENCY DISTRIBUTION IN THE PRINTED OUTPUT IS OVER 6 INTERVALS OF EQUAL LENGTH. THE 1ST AND 6TH INTERVALS ARE CENTERED ON THE LEAST AND GREATEST VALUES OCCURRING IN V.

(4) <u>STRES</u>. NO ARGUMENTS; SHOULD BE USED ONLY AFTER 'REGR' HAS BEEN EXECUTED.

GLOBAL OUTPUT VARIABLES:

DIAGQ IS A VECTOR CONSISTING OF THE MAIN DIAGONAL OF A MATRIX Q (NOT FURTHER DETERMINED HERE) THAT PROJECTS Y INTO RY, THUS RY ↔ Y+.×Q .

SRY EACH ELEMENT OF RY IS DIVIDED BY THE SQUARE ROOT OF THE CORRESPONDING ELEMENT OF DIAGQ (TIMES A CONSTANT). IF THE MEMBERS OF DIAGQ ARE ALL EQUAL, SRY IS THE SAME AS RY.

(5) <u>VARIANCE</u>. NO ARGUMENTS; USE ONLY AFTER 'REGR' HAS BEEN EXECUTED.

ONE GLOBAL OUTPUT VARIABLE:

XPXI IS 'X-PRIME X INVERSE'. WHEN REGRESSION HAS BEEN PERFORMED ON ALL COLUMNS OF X, XPXI IS THE INVERSE OF (⍉X)+.×X; OTHERWISE IT IS THE CORRESPONDING EXPRESSION FOR THE COLUMNS OF X THAT HAVE BEEN USED, TOGETHER WITH ZEROS FOR REGRESSION COEFFICIENTS NOT ESTIMATED.

27 JUNE 1973

```
      ∇ X REGRINIT Y;J
[1]     →3-(∧/1≤ρX)∧(∧/1≤ρY)∧(∨/ 1 2 =ρρY)∧2=ρρX
[2]     →3+J←(¯1↑ρY)=NU←1↑ρX
[3]     →0,ρ⎕←'NO GO.'
[4]     X←0↑,TRX←(P,P)ρ1,IND←(P←¯1↑ρRX←X)ρ0
[5]     →9×ι1=ρρISS←+/RY×RY←Y
[6]     RB←Pρ0
[7]     C̲,'INITIAL SUM OF SQUARES (ISS), NUMBER OF READINGS AND MEAN SQUARE ARE'
[8]     →0,⎕←(⌊0.5+ISS),NU,0.001×⌊0.5+1000×ISS÷NU
[9]     RB←((1↑ρY),P)ρ0
[10]    C̲,'    *Y-VARIABLE ';J;'*'
[11]    'INITIAL SUM OF SQUARES (ISS[';J;']), NUMBER OF READINGS AND MEAN SQUARE A
        RE'
[12]    (⌊0.5+ISS[J]),NU,0.001×⌊0.5+1000×ISS[J]÷NU
[13]    →10×(1↑ρY)≥J←J+1
      ∇
```

```
      ∇ STRES;M
[1]     →2+∧/0<DIAGQ←1-+/M÷(ρM)ρ+⌿M←(IND/RX)*2
[2]     →0,ρ⎕←'A MEMBER OF DIAGQ VANISHES.  SRY NOT FOUND.'
[3]     SRY←RY×(ρRY)ρ(DIAGQ×(ρRX)[1]÷NU)*¯0.5
      ∇
```

```
    ∇ REGR L;D;I;J;K;S;U
[1]   →2+(0<D←4)∧J←(∧/L∊⍳P)∧∧/1≤⍴L←,L
[2]   →0,⍴⍞←'NO GO.'
[3]   →4+1≤NU←NU-0=IND[K←L[J]]
[4]   →0,⍴⍞←'NU HAS VANISHED.'
[5]   IND[K]←1
[6]   →9×⍳(NU÷D)≤S←RX[;K]+.×RX[;K]
[7]   'DIVISOR TOO SMALL AT COLUMN ';K;' OF RX.'
[8]   →0
[9]   RX←RX-RX[;K]∘.×U←(RX[;K]+.×RX×(⍴RX)⍴~IND)÷S
[10]  TRX←TRX-U∘.×TRX[K;]
[11]  →12+2=⍴⍴RY←RY-(U←(RY+.×RX[;K])÷S)∘.×RX[;K]
[12]  →14,RB[K]←RB[K]+U
[13]  RB[;K]←RB[;K]+U
[14]  →3×⍳(⍴L)≥J←J+1
[15]  I←(NU÷D)≤S←+⌿RX×RX
[16]  RSS←+/RY×RY
[17]  →22×⍳J←∨/2=⍴⍴B←RB+.×TRX
[18]  C̲,'RESIDUAL SUM OF SQUARES (RSS), D.F. (NU) AND MEAN SQUARE ARE'
[19]  (⌊0.5+RSS),NU,0.001×⌊0.5+1000×RSS÷NU
[20]  'POSSIBLE REDUCTIONS IN  RSS  ARE'
[21]  →28,⍞←⌊0.5+((I×RY+.×RX)*2)÷S⌈NU÷D
[22]  C̲,'    *Y-VARIABLE ';J;'*'
[23]  'RESIDUAL SUM OF SQUARES (RSS[';J;']), D.F. (NU) AND MEAN SQUARE ARE'
[24]  (⌊0.5+RSS[J]),NU,0.001×⌊0.5+1000×RSS[J]÷NU
[25]  'POSSIBLE REDUCTIONS IN  RSS[';J;']  ARE'
[26]  ⌊0.5+((I×RY[J;]+.×RX)*2)÷S⌈NU÷D
[27]  →22×⍳(1↑⍴RY)≥J←J+1
[28]  →0×⍳1=∧/I
[29]  C̲,'DIVISOR TOO SMALL AT COLUMN ';(~I)/⍳P;' OF RX.'
    ∇


    ∇ SHOW V;S;R;W
[1]   →3-(∨/6≤⍴V)∧1=⍴⍴V
[2]   →3+0<-/V[(R←(⍋V)[(⍳3),(1+⍴V)-⌽⍳3])[6 1]]
[3]   →0,⍴⎕←'NO GO.'
[4]   S←0.01×⌊0.5+100×((+/V×V)÷NU)*0.5
[5]   W←0.01×⌊0.5+100×V[R]
[6]   '3 LOWEST, 3 HIGHEST (WITH RANKS) AND R.M.S. VALUES'
[7]   1↑W;' (';1↑R;'), ';W[2];' (';R[2];'), ';W[3];' (';R[3];'), ';W[4];' (';R[
      4];'), ';W[5];' (';R[5];'). ';5↓W;' (';5↓R;'); ';S
[8]   'FREQUENCY DISTRIBUTION:  ';+/(0,⍳5)∘.=⌊0.5+5×(V-V[1↑R])÷-/V[R[6
      1]]
    ∇


    ∇ VARIANCE
[1]   XPXI←(((P,P)⍴IND\÷+⌿IND/RX×RX)×⍉TRX)+.×TRX
[2]   →(1 2 =⍴⍴RY)/ 3 4
[3]   →0,⍴⎕←'ESTIMATED VARIANCE MATRIX OF B IS:  XPXI×RSS÷NU'
[4]   'ESTIMATED VARIANCE MATRIX OF B[J;] IS:  XPXI×RSS[J]÷NU'
    ∇
```

```
    ∇ ROWCOL Y;N;A
[1]   →2+(6≤×/ρY)∧(∧/2≤ρY)∧2=ρρY
[2]   →0,ρ⎕←'NO GO.'
[3]   ⍞,'GRAND MEAN (GM) IS ';GM←(+/,Y)÷N←×/ρY
[4]   'TOTAL SUM OF SQUARES ABOUT MEAN (TSS), DEGREES OF FREEDOM AND MEAN SQUARE
       ARE ';TSS,(N-1),(÷N-1)×TSS←+/,RY×RY←Y-GM
[5]   ⍞,'    *ROWS*'
[6]   'EFFECTS (RE) ARE ';RE←(+/RY)÷1↓ρY
[7]   'SUM OF SQUARES (SSR), D.F. AND M.S. ARE ';SSR,(¯1+1↑ρY),(÷¯1+1↑ρY)×SSR←(1
      ↓ρY)×RE+.×RE
[8]   ⍞,'    *COLUMNS*'
[9]   'EFFECTS (CE) ARE ';CE←(+⌿RY)÷1↑ρY
[10]  'SUM OF SQUARES (SSC), D.F. AND M.S. ARE ';SSC,(¯1+1↓ρY),(÷¯1+1↓ρY)×SSC←(1
      ↑ρY)×CE+.×CE
[11]  ⍞,'    *RESIDUALS*'
[12]  'SUM OF SQUARES (RSS), D.F. (NU) AND M.S. ARE ';RSS,NU,(÷NU←×/¯1+ρY)×RSS←+
      /,RY×RY←RY-RE∘.+CE
[13]  'THE MATRIX OF RESIDUALS IS NAMED RY.'
[14]  →(0<SSR×SSC)/¯2+ρ⎕←⍞,'    *TUKEY''S TEST*'
[15]  →ρ⎕←'TEST FAILS BECAUSE  SSR×SSC  VANISHES.'
[16]  'REGRESSION COEFFT. (B) OF RY ON RE∘.×CE IS ';B←(A←+/,RY×RE∘.×CE)×N÷SSR×
      SSC
[17]  'TUKEY''S 1-D.F. TERM IS ';A←A×B
[18]  'REMAINING SUM OF SQUARES, D.F. AND M.S. ARE ';A,(NU-1),(÷NU-1)×A←RSS-A
[19]  ⍝  THIS PROGRAM PERFORMS A STANDARD ANALYSIS OF VARIANCE ON A TWO-WAY TABL
      E.
    ∇


    ∇ ROWCOLPERMUTE
[1]   RE←RE[RP←⍒RE]
[2]   CE←CE[CP←⍋CE]
[3]   RY←RY[RP;CP]
[4]   'ROW PERMUTATION (RP) IS ';RP
[5]   'COLUMN PERMUTATION (CP) IS ';CP
[6]   'RE, CE AND RY ARE NOW PERMUTED.'
[7]   ⍝  USE AFTER ROWCOL, BEFORE ROWCOLDISPLAY.
    ∇


    ∇ ROWCOLDISPLAY I;C;D;R;S;W
[1]   →2+(0<+/,I)∧(0=ρρI)∧(2=ρS← 1 1)∧(D←2)∊⍳2
[2]   →0,ρ⎕←'NO GO.'
[3]   →4+130≥ρC←(¯1↑W←(D,D×1⌈⌊0.5+((1↓CE)-¯1↓CE)÷I)+.×(⍳ρCE)∘.≤⍳ρCE)ρ0
[4]   →ρ⎕←'THE ARGUMENT IS TOO SMALL, TRY AGAIN.'
[5]   C[W]←1
[6]   ⍞,'    *CODED DISPLAY OF PERMUTED RESIDUALS*'
[7]   R←(¯1↑W←(1,1⌈⌊0.5+((¯1↓RE)-1↓RE)÷÷/I,S)+.×(⍳ρRE)∘.≤⍳ρRE)ρ0
[8]   R[W]←1
[9]   'NE-SW DISPLACEMENTS ROUGHLY MEASURE CHANGES IN FITTED VALUES.',⍞
[10]  C\R⍀'MM-∘+PP'[⌊4.5+3×RY÷⌈/,|RY]
[11]  ⍝  SEE 'ROWTRIPLEPLOT'.
    ∇
```

```
    ∇ ANALYZE Y
[1]   →2+(∧/2≤ρY)∧2≤ρρY
[2]   →0,ρ⎕←'NO GO.'
[3]   DF←(1,1+ρρY)ρ1
[4]   'GRAND MEAN (GM) IS ';GM←(+/,Y)÷×/ρY
[5]   'TOTAL SUM OF SQUARES ABOUT MEAN (TSS), DEGREES OF FREEDOM AND MEAN SQUARE
       ARE ';TSS,NU,(÷NU←¯1+×/ρY)×TSS←+/,RY×RY←Y-GM
[6]   'PROCEED BY REPEATEDLY CALLING THE FUNCTION ''EFFECT''.'
[7]   'THE ARRAY OF RESIDUALS IS ALWAYS NAMED RY, WITH DEGREES OF FREEDOM NU.'
[8]   'RENAME SS EACH TIME TO SAVE IT.'
[9]   ⍝ BEGINS AN ANALYSIS OF VARIANCE OF A PERFECT RECTANGULAR ARRAY.
    ∇


    ∇ Z←EFFECT V;J;K;M;P;R
[1]   →3-∧/∨/M←(,V)∘.=⍳R←+/ρρRY
[2]   →3+∧/0≤M←1-+⌿M
[3]   →0,ρ⎕←'NO GO.',Z←''
[4]   J←+/1↑ρDF←DF,[1] M,0
[5]   NU←NU-DF[J;1+R]←(×/(ρRY)[V])-DF[;1+R]+.×(+/DF[J;])=DF+.×DF[J;]
[6]   SS←K×+/,Z×Z←(÷K)×+⌿((K←×/M/ρRY),(ρRY)[V])ρ(⍋P←(M/⍳R),V)⍉RY
[7]   RY←RY-P⍉((M/ρRY),(ρRY)[V])ρZ
[8]   'SUM OF SQUARES (SS), D.F. AND M.S. ARE ';SS,DF[J;1+R],SS÷DF[J;1+R]
[9]   ⍝ 'ANALYZE' SHOULD BE CALLED FIRST.
    ∇


    ∇ Y←STDIZE X;C
[1]   →2+∨/X≠1↑X←,X
[2]   →0,ρ⎕←'NO GO.',Y←''
[3]   Y←Y×((÷¯1+ρX)×+/Y×Y←X-(+/X)÷ρX)*¯0.5
[4]   ⍝ THE VECTOR X IS RESCALED TO HAVE ZERO MEAN AND UNIT VARIANCE.
    ∇
```

HOWFILTER

FILTERING TIME SERIES
(1) U←M FILTER X
(2) U←M MAV X

(1) <u>FILTER</u>. THE FIRST ARGUMENT (M) IS A PAIR OF INTEGERS DEFINING THE FILTER. THE SECOND ARGUMENT (X) IS A VECTOR OF DATA TO BE FILTERED. FILTERING CONSISTS OF SUBTRACTING A COSINE-WEIGHTED MOVING AVERAGE OF EXTENT M[2] FROM A SIMILAR MOVING AVERAGE OF EXTENT M[1] . THE RESULTING WEIGHTS ARE DISPLAYED, TOGETHER WITH THEIR SUM OF SQUARES. THE ELEMENTS OF M MUST BE EITHER BOTH ODD OR BOTH EVEN, OR ELSE ONE OF THEM MUST BE 0. TO GET AN 11-POINT MOVING AVERAGE, SET M EQUAL TO 11 0 . TO DETREND A SERIES BY SUBTRACTING FROM IT AN 11-POINT MOVING AVERAGE, LET M BE 1 11 . FOR A SMOOTHED DETRENDED SERIES, LET M BE (E.G.) 3 11 . SETTING M EQUAL TO 3 1 YIELDS THE SECOND DIFFERENCE OF X (DIVIDED BY 4). IF THE NONZERO ELEMENTS OF M ARE ODD, THE RESULT CORRESPONDS TO THE SAME TIME POINTS AS THE DATA (APART FROM LOSS OF SOME END VALUES). BUT IF BOTH ELEMENTS OF M ARE EVEN, THE RESULT CORRESPONDS TO TIME POINTS MIDWAY BETWEEN THOSE OF THE DATA.

IF AN ELEMENT OF M EXCEEDS 4, A LITTLE CHEATING GOES ON AT THE ENDS OF X , THE FIRST AND LAST ELEMENTS BEING REPEATED R TIMES AS DEFINED AT [7]. IF YOU DO NOT WANT THIS FEATURE, EITHER REPLACE [7] BY
[7] R←0
OR DROP THE FIRST AND LAST R ELEMENTS OF U .

(2) <u>MAV</u>. AN ADAPTATION OF 'FILTER' TO TAKING A COSINE-WEIGHTED MOVING AVERAGE OF EACH OF SEVERAL VECTORS SIMULTANEOUSLY. THE FIRST ARGUMENT (M) IS A POSITIVE INTEGER, THE SECOND (X) IS A MATRIX. THE RESULT (U) IS A MATRIX, WHOSE ROWS ARE WHAT WOULD BE OBTAINED BY REPEATED APPLICATION OF 'FILTER' (MODIFIED AS SUGGESTED ABOVE BY SETTING R←0) TO THE ROWS OF X :
U[J;] ↔ (M,0) FILTER X[J;]
IF YOU WANT THE WEIGHTS TO BE DISPLAYED, INSERT + AT THE BEGINNING OF LINE [4] OR SOMETHING.

28 JUNE 1973

```
    ∇ U←M FILTER X;P;R;W
[1]   →3-(∧/2=ρM)∧(1=ρρM)∧1=ρρX
[2]   →3+∧/(0≤M)∧(M=⌊M)∧((0=×/M)∨0=2|-/M)∧(≠/M)∧(2×¯2+ρX)≥P←⌈/M
[3]   →0,ρ⎕←'NO GO.',U←''
[4]   W←(⌊0.5×P-M[1])⌽((P-M[1])ρ0),(1-2○(⍳M[1])×○2÷1+M[1])÷1+M[1]
[5]   W←W-(⌊0.5×P-M[2])⌽((P-M[2])ρ0),(1-2○(⍳M[2])×○2÷1+M[2])÷1+M[2]
[6]   'FILTER WEIGHTS ARE  ';W;'  WITH S.S.  ';W+.×W
[7]   R←⌊(P-1)÷4
[8]   →L1×⍳(⌶22)≤16×P×ρX←(Rρ1↑X),X,Rρ¯1↑X
[9]   →0,ρU←W+.×(0,1-P)↓(¯1+⍳P)⌽(P,ρX)ρX
[10] L1:R←ρρU←((ρX)+1-P)ρ0
[11]  U←U+W[R]×(R-1)↓(R-P)↓X
[12]  →(P≥R←R+1)/L1+1
    ∇
```

HOWHARMONIC

FOURIER ANALYSIS OF A TIME SERIES
(1) *V←K TAPER U*
(2) *Z←HA V*

(1) *TAPER*. *THE FIRST ARGUMENT (K) IS A POSITIVE INTEGER, AND THE SECOND (U) IS A VECTOR OF LENGTH GREATER THAN 2×K . THE RESULT (V) IS A VECTOR OF LENGTH (ρU)-K , A CIRCULARIZED VERSION OF U OBTAINED BY OVERLAPPING THE FIRST AND THE LAST K ELEMENTS. THE FIRST K ELEMENTS OF U MULTIPLIED BY A VECTOR OF LINEAR WEIGHTS W , PLUS THE LAST K ELEMENTS OF U MULTIPLIED BY 1-W , FORM K ELEMENTS OF V , SOME AT THE BEGINNING AND SOME AT THE END. THE REST OF V IS THE SAME AS THE REST OF U . THE WEIGHTS W ARE DISPLAYED. DIFFERENT WEIGHTS CAN BE OBTAINED BY ALTERING THE DEFINITION IN LINE [4]. THIS PROCESS IS NEARLY BUT NOT QUITE WHAT IS GENERALLY CALLED TAPERING. TO PERFORM A HARMONIC ANALYSIS OF A GIVEN TIME SERIES, ONE MAY SUCCESSIVELY EXECUTE 'FILTER', 'TAPER' AND 'HA'.*

(2) *HA*. *THE ARGUMENT (V) IS A VECTOR OF LENGTH N NOT LESS THAN 3. THE RESULT (Z) IS A MATRIX WITH TWO ROWS. THE FIRST ROW IS THE FOURIER DECOMPOSITION OF THE TOTAL SUM OF SQUARES OF V (ESSENTIALLY THE LINE SPECTRUM OR PERIODOGRAM). THE SECOND ROW LISTS PHASES. THE ANALYSIS OF VARIANCE SHOULD BE CHECKED FOR NUMERICAL ACCURACY:*

```
+/Z[1;] ↔ V+.×V
```

Z[1;1] *IS THE 'CORRECTION FOR THE MEAN' WITH 1 DEGREE OF FREEDOM. FOR NUMERICAL STABILITY IT OUGHT TO BE SMALL. THE CORRESPONDING PHASE* Z[2;1] *IS SET CONVENTIONALLY EQUAL TO* 0. *IF N IS EVEN, THE LAST MEMBER OF* Z[1;] *ALSO HAS* 1 *D.F. AND THE CORRESPONDING PHASE* ¯1↑Z[2;] *IS SET CONVENTIONALLY EQUAL TO* 0. *ALL OTHER MEMBERS OF* Z[1;] *HAVE* 2 *D.F.* Z[1;J+1] *REFERS TO THE J -TH HARMONIC, J CYCLES PER N TIME UNITS, WHERE J IS LESS THAN N÷2 . PHASE (BETWEEN* 0 *AND* ○2) *IS MEASURED RELATIVE TO A TIME ORIGIN ON THE LEFT, SO THAT THE OBSERVED TIME POINTS ARE* ⍳N . *THE FITTED HARMONIC IS*

```
AMPLITUDE × COSINE (PHASE + (○2×J÷N)×⍳N)
```

WHERE

```
AMPLITUDE ← (Z[1;J+1]×2÷N)*0.5
PHASE ← Z[2;J+1]
```

15 *NOV.* 1971

```
    ∇ Z←HA V;J;M;N;A
[1]   →2+(3≤N←+/ρV)∧(J←1)=ρρV
[2]   →0,ρ⍞←'NO GO.',Z←''
[3]   Z←(2,1+⌊N÷2)ρ((A×A←+/V)÷N),(N+1)ρ0
[4]   M← 1 2 ∘.○(○2÷N)×⍳N
[5]   Z[1;J+1]←(2÷N)×A+.×A←M[;1+N|¯1+J×⍳N]+.×V
[6]   →8-0=A[2]
[7]   →9,Z[2;J+1]←○0.5+1=×A[1]
[8]   Z[2;J+1]←(○2)|(○¯1=×A[2])-¯3○÷/A
[9]   →(1 0 ¯1 =×N-2×J←J+1)/ 5 10 0
[10]  Z[1;J+1]←(A×A←-/V)÷N
    ∇
```

```
    ∇ U←M MAV X;R;W
[1]   →3-(0=ρρM)∧2=ρρX
[2]   →3+(M≥1)∧(M=⌊M)∧M≤¯1↑ρX
[3]   →0,ρ⎕←'MAV NO GO.',U←''
[4]   W←(1-2○(⍳M)×○2÷1+M)÷1+M
[5]   M;'-POINT COSINE-WEIGHTED MOVING AVERAGES'
[6]   R←ρρρU←((ρX)-0,M-1)ρ0
[7]  L1:U←U+W[R]×(0,R-1)↓(0,R-M)↓X
[8]   →(M≥R←R+1)/L1
    ∇


    ∇ V←K TAPER U;W
[1]   →3-(0=ρρK)∧1=ρρU
[2]   →3+(K=⌊K)∧(K≥1)∧1≤(ρU)-2×K
[3]   →0,ρ⎕←'NO GO.',V←''
[4]   'TAPER WEIGHTS ARE  ';W←(⍳K)÷K+1
[5]   U[⍳K]←((-K)↑U)+W×U[⍳K]-(-K)↑U
[6]   V←(⌊K÷2)⌽(-K)↓U
    ∇


    ∇ C←N AUTOCOV V;J
[1]   →2+(N≤¯1++/ρV)∧(J←2≤+/ρV)∧(1=ρρV)∧0=ρρN
[2]   →0,ρ⎕←'NO GO.',C←''
[3]   C←V+.×V
[4]   →4×N≥J←ρC←C,(J↓V)+.×(-J)↓V
[5]   ⍝  FIRST N TERMS OF CORRELOGRAM OF V ARE  1↓C÷C[1]
    ∇


    ∇ Z←NIF X
[1]   Z←0.5+(×X)×0.5-(*X×X÷¯2)×((÷1+0.2316419×|X)∘.*⍳5)+.× 319381530 ¯356563782
      1781477937 ¯1821255978 1330274429 ×3.989422804E¯10
[2]   ⍝  HASTINGS APPROXIMATION TO THE NORMAL (GAUSS-LAPLACE) INTEGRAL FUNCTION,
       NBS HANDBOOK, AMS55, 26.2.17.
[3]   ⍝  ABSOLUTE ERROR IN Z LESS THAN 1E¯7.  X MAY BE ANY NUMERICAL ARRAY.
    ∇


    ∇ X←INIF P;U
[1]   →2+∧/,(1>P)∧0<P
[2]   →ρ⎕←'NO GO.',X←''
[3]   U←(¯2×⍟0.5-|P-0.5)*0.5
[4]   X←(×P-0.5)×U-((U∘.*¯1+⍳3)+.× 2515517 802853 10328)÷(U∘.*¯1+⍳4)+.×
      1000000 1432788 189269 1308
[5]   ⍝  HASTINGS APPROXIMATION TO THE INVERSE OF THE NORMAL INTEGRAL FUNCTION,
      NBS HANDBOOK, AMS55, 26.2.23.
[6]   ⍝  ABSOLUTE ERROR IN X LESS THAN 5E¯4.  P MAY BE ANY ARRAY OF NUMBERS BETW
      EEN 0 AND 1.
    ∇
```

HOWHUBER

ROBUST REGRESSION
(1) ZZ←R HUBER Z
(2) DZ←HUBER1
(3) DZ←HUBER2

THE FUNCTION 'HUBER' PERFORMS ONE CYCLE OF ITERATION TOWARDS MINIMIZING THE SUM OF A FUNCTION RHO OF THE RESIDUALS IN A REGRESSION PROBLEM, WHERE RHO IS DEFINED IN TERMS OF A POSITIVE CONSTANT K AS FOLLOWS:

```
      ∇ U ← RHO Z
[1]   U←((0.5×Z*2)×K≥|Z)+(K×(|Z)-K÷2)×K<|Z  ∇
```

(SEE P. J. HUBER: ANN. MATH. STATIST. 35 (1964), 73-101.)

K MAY BE SPECIFIED IN ADVANCE, OR ALTERNATIVELY MAY BE DETERMINED BY THE CONDITION THAT A SPECIFIED PROPORTION OF RESIDUALS EXCEED K IN MAGNITUDE. CALCULATION OF ESTIMATED CHANGES IN THE REGRESSION PARAMETERS AND IN THE RESIDUALS IS DONE BY A SUBSIDIARY FUNCTION, 'HUBER1' OR 'HUBER2'. IF NEITHER OF THESE FUNCTIONS IS APPROPRIATE FOR THE DATA, INSERT A NEW LINE IN 'HUBER' AS FOLLOWS:

```
[8.1] →L4,ρDZ←HUBER3
```

AND DEFINE A NEW SUBSIDIARY FUNCTION STARTING

```
      ∇ DZ←HUBER3; . . . ∇
```

(1) HUBER. THE FIRST ARGUMENT (R) MUST BE SCALAR. IF IT IS ¯1, THE CURRENT GLOBAL VALUE FOR K IS USED; OTHERWISE R MUST LIE BETWEEN 0 AND 1, AND IS THE PROPORTION OF RESIDUALS TO EXCEED K IN MAGNITUDE. THE SECOND ARGUMENT (Z) IS THE ARRAY OF RESIDUALS CORRESPONDING TO A TRIAL SETTING OF THE REGRESSION PARAMETERS BETA , WHICH MUST BE SPECIFIED BEFORE 'HUBER' IS CALLED. THE EXPLICIT RESULT (ZZ) IS A NEW SET OF RESIDUALS, THAT CAN BE USED AS SECOND ARGUMENT IN THE NEXT CALL OF 'HUBER'.

GLOBAL OUTPUT VARIABLES: K, DBETA (ESTIMATED REQUIRED CHANGE IN PARAMETERS), BETA (NEW PARAMETER VALUES).

THE PARAMETER CHANGE DBETA YIELDED BY THE SUBSIDIARY FUNCTION IS DETERMINED FROM THE FIRST AND SECOND DERIVATIVES OF THE SUM OF RHOS AT THE INITIAL BETA . IT YIELDS THE TRUE MINIMUM OF THE SUM OF RHOS, FOR THE VALUE OF K USED, IF NO CHANGE OCCURS IN THE INDEXING OF THE RESIDUALS THAT EXCEED K IN MAGNITUDE -- THOSE RESIDUALS BEING REFERRED TO AS 'MODIFIED RESIDUALS'. OTHERWISE THE TRUE MINIMUM IS NOT OBTAINED. THEN THE RESIDUALS ARE CALCULATED WITH THE FULL PARAMETER CHANGE DBETA AND ALSO WITH CHANGES THAT ARE ONLY 0.9, 0.8,... OF DBETA, STOPPING WHEN THE SUM OF RHOS IS LEAST. THE MULTIPLIER OF DBETA IS NAMED ALPHA IN THE DISPLAYED OUTPUT. THE VALUE ALPHA = 0.0 CORRESPONDS TO THE INPUT PARAMETER SETTING BETA . WHAT IS FINALLY YIELDED AS THE OUTPUT VARIABLE DBETA IS THE FIRST DBETA MULTIPLIED BY THE BEST ALPHA ; AND THAT DBETA IS ADDED TO THE INPUT BETA TO MAKE THE OUTPUT BETA .

'HUBER1' AND 'HUBER2' WILL FAIL IN EXECUTION IF THE RIGHT ARGUMENT OF ⌹ IS SINGULAR. IF THAT HAPPENS, (CLEAR THE STATE INDICATOR AND) START 'HUBER' AGAIN WITH A LARGER K (OR SMALLER POSITIVE R).

(2) HUBER1 IS INVOKED WHEN Z IS A VECTOR. THERE MUST BE A GLOBAL MATRIX X WHOSE COLUMNS ARE THE INDEPENDENT VARIABLES IN A REGRESSION. FOR CONSISTENCY

```
      ρZ ↔ 1↑ρX ;  ρBETA ↔ ¯1↑ρX .
```

THE ORIGINAL DATA VECTOR, NOT NAMED IN THE PROGRAM, WAS EQUAL INITIALLY TO

```
      Z + X+.×BETA .
```

(3) HUBER2 IS INVOKED WHEN Z IS A MATRIX. Z MUST HAVE AT LEAST 3 ROWS AND 3 COLUMNS. THE USUAL ADDITIVE STRUCTURE IS FITTED. BETA IS A VECTOR OF LENGTH 1++/ρZ , AND LISTS THE MEAN, THE ROW CONSTANTS (SUMMING TO 0), AND THE COLUMN CONSTANTS (SUMMING TO 0). THE ORIGINAL DATA MATRIX, NOT NAMED IN THE PROGRAM, WAS EQUAL INITIALLY TO

```
        Z + BETA[1] + ((ρZ)[1]↑1↓BETA)∘.+(ρZ)[1]↓1↓BETA .
```

23 JUNE 1973

```
     ∇ ZZ←R HUBER Z;DZ;I;I1;J;N;P;S;S1;ZZ1
[1]    →2+(ρρBETA←,BETA)∧(0=ρρR)∧J←3≤N←×/ρZ
[2]    →L1,ρ⎕←'HUBER NO GO.'
[3]    →(R=¯1)/6
[4]    →((1>P)∨N<P←0.5+N×1-R)/2
[5]    K←+/(1 0 + ¯1 1 ×P-⌊P)×|(,Z)[(⍋|,Z)[(⌊P),⌈P]]
[6]    'K = ';K
[7]    'INITIAL NUMBER OF MODIFIED RESIDUALS = ';+/,I←K<|Z
[8]    'FOR ALPHA = 0.0, SUM OF RHOS = ';(0.5×+/,Z×Z×~I)+K×+/,I×(|Z)-K÷2
[9]    →((1 2 =ρρZ)/L2,L3),2
[10] L1:→0,ZZ←''
[11] L2:→(L1×⍳0=×/ρDZ←HUBER1),L4
[12] L3:→L1×⍳0=×/ρDZ←HUBER2
[13] L4:→L6×⍳∧/,I=I1←K<|ZZ←Z+DZ
[14]   'FOR ALPHA = 1.0, SUM OF RHOS = ';S←(0.5×+/,ZZ×ZZ×~I1)+K×+/,I1×(|ZZ)-K÷
       2
[15] L5:I1←K<|ZZ1←Z+DZ×J←J-0.1
[16]   'FOR ALPHA = ';J;', SUM OF RHOS = ';S1←(0.5×+/,ZZ1×ZZ1×~I1)+K×+/,I1×(|ZZ1)
       -K÷2
[17]   →L7×⍳(S≤S1)∨J=0
[18]   S←S1
[19]   →L5,ρZZ←ZZ1
[20] L6:'MINIMIZED SUM OF RHOS = ';(0.5×+/,ZZ×ZZ×~I)+K×+/,I×(|ZZ)-K÷2
[21] L7:'DBETA = ';DBETA←DBETA×1⌊J+0.1
[22]   'BETA = ';BETA←BETA+DBETA
     ∇
```

```
     ∇ DZ←HUBER1;Y
[1]    →2+((ρBETA)=¯1↑ρX)∧(N=1↑ρX)∧(2=ρρX)∧1=ρρZ
[2]    →ρ⎕←'HUBER1 NO GO.',DZ←''
[3]    DZ←-X+.×DBETA←((K⌊Z⌈-K)+.×X)⌹(⍉Y)+.×Y←(~I)⌿X
     ∇
```

```
     ∇ DZ←HUBER2;A;B;C;D;M;U
[1]    →2+((ρBETA)=1++/ρZ)∧(∧/3≤ρZ)∧2=ρρZ
[2]    →ρ⎕←'HUBER2 NO GO.',DZ←''
[3]    U←(+/,U),(+/U),+⌿U←K⌊Z⌈-K
[4]    C←((ρZ)[2 2]ρ1,(ρZ)[2]ρ0)×(ρZ)[2 2]ρA←+⌿~I
[5]    D←((ρZ)[1 1]ρ1,(ρZ)[1]ρ0)×(ρZ)[1 1]ρB←+/~I
[6]    M←((+/B),-(+/I),+⌿I),[1](B,D,-I),[1] A,(⍉-I),C
[7]    A←B←C←D←''
[8]    DZ←-(1↑DBETA)+((ρZ)[1]↑1↓DBETA)∘.+(ρZ)[1]↓1↓DBETA←U⌹M
     ∇
```

HOWCONTINGENCY

ANALYSIS OF CONTINGENCY TABLES
(1) CONTINGENCY M
(2) FOURFOLD M
(3) MULTIPOLY M
(4) Y←V POOL X

THE FIRST TWO FUNCTIONS APPLY TO 2-DIMENSIONAL CONTINGENCY TABLES, THE OTHERS TO CONTINGENCY TABLES IN ANY NUMBER OF DIMENSIONS. 'CONTINGENCY' PERFORMS A CHI SQUARED TEST OF ASSOCIATION, WITH DISPLAY OF STANDARDIZED RESIDUALS. 'FOURFOLD' MAY BE APPLIED WHEN THE CATEGORIES OF EACH CLASSIFICATION ARE ORDERED. EMPIRICAL LOG CROSSPRODUCT (FOURFOLD) RATIOS ARE DISPLAYED; A PLACKETT DISTRIBUTION IS FITTED, AND GOODNESS OF FIT IS TESTED BY CHI SQUARED, WITH DISPLAY OF STANDARDIZED RESIDUALS. 'MULTIPOLY' FINDS EMPIRICAL LOG CROSSPRODUCT RATIOS, ANALOGOUS TO THOSE OF 'FOURFOLD' FOR 2-DIMENSIONAL TABLES. 'POOL' POOLS CATEGORIES IN ANY TABLE.

(1) <u>CONTINGENCY</u>. THE ARGUMENT (M) MUST BE A MATRIX OF NONNEGATIVE INTEGERS, HAVING NO ZERO MARGINAL TOTAL. (IF M HAS FRACTIONAL COUNTS, LINE [1] SHOULD BE BYPASSED.) GLOBAL OUTPUT VARIABLES: EF (MATRIX OF EXPECTED FREQUENCIES), SR (MATRIX OF STANDARDIZED RESIDUALS). SR IS DEFINED THUS:

```
        SR ← (M-EF)÷EF*0.5
+/,SR×SR  IS CHI SQUARED.
```

<u>NOTE</u>. FOR GUTTMAN PREDICTION OF THE COLUMN CATEGORY FROM THE ROW CATEGORY,

```
    LAMBDA ← ((+/⌈/M)-G)÷(+/,M)-G←⌈/+⌿M
```

(SEE GOODMAN AND KRUSKAL: <u>JASA</u> 49 (1954), 732-764.)

(2) <u>FOURFOLD</u>. SAME REQUIREMENT FOR M AS IN 'CONTINGENCY'. THE MATRIX OF LOG FOURFOLD RATIOS HAS DIMENSION VECTOR (⍴M)-1 , AND REFERS TO ALL POSSIBLE 2-BY-2 TABLES THAT CAN BE FORMED FROM M BY POOLING ADJACENT ROWS AND COLUMNS. IF

```
        N11 | N12
        ---------
        N21 | N22
```

IS SUCH A TABLE, THE NATURAL-LOG FOURFOLD RATIO (LFR) CALCULATED IS

```
        ⍟÷/0.5+(N11,N12,N22,N21)
```

THE 0.5 ADDED TO EACH COUNT MAKES LFR A ROUGHLY UNBIASED ESTIMATE OF LOG PSI, WHERE PSI IS THE CORRESPONDING FOURFOLD RATIO OF PROBABILITIES, AND PREVENTS DISASTER IF ANY COUNT IS ZERO.

POSSIBLY THESE VALUES OF LFR DIFFER ONLY BY SAMPLING ERROR. A PLACKETT DISTRIBUTION HAVING CONSTANT PSI IS FITTED ITERATIVELY BY MAXIMUM CONDITIONAL LIKELIHOOD, GIVEN THAT THE MARGINAL PROBABILITIES ARE PROPORTIONAL TO THE GIVEN MARGINAL TOTALS. AT STAGE 0 THE THREE TRIAL VALUES FOR LOG PSI ARE THE MOST PRECISE OF THE MEMBERS OF LFR AND THE SAME ± ONE ESTIMATED STANDARD ERROR. THE LOG LIKELIHOOD FUNCTION IS COMPUTED FOR THE 3 VALUES OF LOG PSI , A PARABOLA IS FITTED, AND AT STAGE 1 THE NEXT 3 TRIAL VALUES FOR LOG PSI ARE THE VALUE TO MAXIMIZE THE PARABOLA AND A VALUE ON EITHER SIDE, USUALLY CLOSER THAN BEFORE. AFTER AT MOST 4 ITERATIONS, EXPECTED FREQUENCIES (EF) ARE CALCULATED, AND HENCE STANDARDIZED RESIDUALS (SR) AND CHI SQUARED.

THE 4 ITERATIONS WILL NOT FIND THE CORRECT M.L. VALUE FOR LOG PSI IF THAT IS INFINITE. TO ADJUST THE MAXIMUM NUMBER OF ITERATIONS, CHANGE THE '4' IN LINE [1] TO ANY INTEGER NOT LESS THAN 2.

SEE

PLACKETT: <u>JASA</u> 60 (1965), 516-522.
TUKEY: <u>EXPLORATORY DATA ANALYSIS</u>, ADDISON WESLEY, PRELIM. ED. 1971, CHAP. 29X.

(3) MULTIPOLY. THE ARGUMENT (M) MUST BE AN ARRAY OF NONNEGATIVE INTEGERS IN 2 OR MORE DIMENSIONS. GLOBAL OUTPUT VARIABLES: LCR (LOG CROSSPRODUCT RATIOS), ESE (ESTIMATED STANDARD ERRORS). LCR AND ESE HAVE DIMENSION VECTOR (ρM)-1 .

IF M IS A MATRIX, LCR IS THE SAME AS LFR YIELDED BY 'FOURFOLD'. IF M IS 3-DIMENSIONAL, LCR REFERS TO ALL POSSIBLE 2-BY-2-BY-2 TABLES THAT CAN BE FORMED FROM M BY POOLING ADJACENT PLANES, ROWS AND COLUMNS. IF

```
N111 | N112     N211 | N212
N121 | N122     N221 | N222
```

IS SUCH A TABLE, THE NATURAL-LOG CROSSPRODUCT (EIGHTFOLD) RATIO LCR IS

```
⍟÷/0.5+(N111,N112,N122,N121,N221,N211,N212,N222)
```

THE VARIANCE MAY BE ROUGHLY ESTIMATED BY

```
+/÷0.5+(N111,N112, . . . ,N222)
```

THE SQUARE ROOTS ARE PUT OUT AS ESE . SIMILARLY IF M HAS MORE THAN THREE DIMENSIONS.

(4) POOL. THE SECOND ARGUMENT (X) IS AN ARRAY IN 2 OR MORE DIMENSIONS, TYPICALLY A CONTINGENCY TABLE OR A TABLE OF EXPECTED FREQUENCIES. THE FIRST ARGUMENT (V) IS A VECTOR WITH AT LEAST 3 ELEMENTS. V[1] SPECIFIES THE COORDINATE, AND 1↓V SPECIFIES THE INDEX-VALUES, OVER WHICH THERE IS TO BE POOLING. SECTIONS OF X CORRESPONDING TO INDEX-VALUES 2↓V OF COORDINATE V[1] ARE ADDED TO THE SECTION WITH INDEX-VALUE V[2] , AND THEN THE FORMER SECTIONS ARE DELETED. THUS IF X IS A MATRIX WITH 5 ROWS, AND IF V IS 1 4 5 1, POOLING WILL BE OVER THE 1ST COORDINATE (ROWS); THE CONTENTS OF ROWS 4, 5 AND 1 WILL BE ADDED, PLACED IN ROW 4, AND THEN ROWS 5 AND 1 WILL BE DROPPED, SO THAT THE RESULT (Y) HAS 3 ROWS, THE OLD 2ND AND 3RD AND THEN THE SUM OF THE OLD 1ST, 4TH AND 5TH ROWS. IF V HAD BEEN 1 1 4 5 THE SAME RESULT WOULD HAVE BEEN OBTAINED EXCEPT FOR A PERMUTATION OF ROWS.

9 JULY 1973

```
     ∇ CONTINGENCY M;C;R
[1]    →2+(∧/,0≤M)∧(∧/,M=⌊M)∧(∧/2≤ρM)∧2=ρρM
[2]    →ρ⍞←'NO GO, THE ARGUMENT SHOULD BE A MATRIX OF NONNEGATIVE INTEGERS.'
[3]    →4+(∧/1≤C←+⌿M)∧∧/1≤R←+/M
[4]    →ρ⍞←'NO GO, A MARGINAL TOTAL VANISHES.'
[5]    'THE MATRIX OF EXPECTED FREQUENCIES IS NAMED  EF .'
[6]    'LEAST MEMBER OF  EF  =  ';⌊/,EF←R∘.×C÷+/R
[7]    'STANDARDIZED RESIDUALS (SR):'
[8]    0.01×⌊0.5+100×SR←(EF*¯0.5)×M-EF
[9]    'CHI-SQUARED  =  ';+/,SR×SR;⍞←''
[10]   'DEGREES OF FREEDOM  =  ';×/¯1+ρM
     ∇
```

```
     ∇ Y←V POOL X;P
[1]    →3-(∧/(1↑,V)≤ρρX)∧(∧/2≤ρX)∧(2≤ρρX)∧(1=ρρV)∧(3≤+/ρV)∧(∧/,1≤V)∧∧/,V=⌊V
[2]    →3+(∧/(1↓V)≤(ρX)[V[1]])∧(¯1+ρV)=+/,(1↓V)∘.=1↓V
[3]    →0,ρ⍞←'NO GO.',Y←''
[4]    Y←((1↑ρX),×/1↓ρX)ρX←(⍋P←V[1],(V[1]≠⍳ρρX)/⍳ρρX)⍉X
[5]    Y[V[2];]←+⌿Y[1↓V;]
[6]    Y←P⍉((1↑ρY),1↓ρX)ρY←Y[(∧⌿(2↓V)∘.≠⍳1↑ρY)/⍳1↑ρY;]
     ∇
```

```
    ∇ FOURFOLD M;W;R;C;GT;RCT;CCT;S;J;D;L;ITS
[1]   →2+(2≤ITS←4)∧(∧/,M≥J←0)∧(∧/,M=⌊M)∧(∧/2≤ρM)∧2=ρρM
[2]   →ρ⎕←'NO GO, THE ARGUMENT SHOULD BE A NONNEGATIVE INTEGER MATRIX WITH POSIT
      IVE MARGINAL TOTALS.'
[3]   W←((⍳R)∘.≥⍳R←(ρM)[1])+.×M+.×(⍳C)∘.≤⍳C←(ρM)[2]
[4]   GT←W[R;C]
[5]   →2×⍳∨/(RCT,GT)=0,RCT←,((R-1),¯1)↑W
[6]   →2×⍳∨/(CCT,GT)=0,CCT←,(¯1,C-1)↑W
[7]   W← ¯1 ¯1 ↓W
[8]   W←0.5+W,((RCT∘.+(C-1)ρ0)-W),(GT+W-RCT∘.+CCT),[2.5](((R-1)ρ0)∘.+CCT)-W
[9]   'LOG FOURFOLD RATIOS (LFR):'
[10]  (⌊0.5+100×LFR←⍟÷/W)÷100
[11]  LP←(,LFR)[W⍳S←⌊/W←,+/÷W]
[12]  LP←LP+ ¯1 0 1 ×S←S*0.5
[13] L1:'STAGE ';J;':   LOG PSI  =  ';LP←LP+1E¯11×GT×∨/GT>200000000000×|LP
[14]  W←GT+(¯1+PSI←*LP)∘.×RCT∘.+CCT
[15]  W←(W-((W×W)-(4×PSI×PSI-1)∘.×RCT∘.×CCT)*0.5)÷(2×PSI-1)∘.+(ρLFR)ρ0
[16]  W←W-0,[2] 0 ¯1 0 ↓W←W-0, 0 0 ¯1 ↓W←(W,[2]((ρLP)ρ0)∘.+CCT),((ρLP)ρ0)∘.+RCT,
      GT
[17]  →L3×⍳J=ITS
[18]  L←+/+/((ρW)ρM)×⍟W
[19]  →L2×⍳0≥D←-/L[2 1 2 3]
[20]  LP←,LP[2]+D←(S×-/L[3 1])÷2×D
[21]  →L1×⍳ITS=J←J+1
[22]  ITS←(1 0 =S≥2×|D)/(J+1),ITS
[23]  →L1,ρLP←LP+ ¯1 0 1 ×S←(S÷4)⌈(0.5×|D)⌊S
[24] L2:'LIKELIHOOD FN CURVES WRONG WAY.  ENTER 3 INCREASING EQUAL-SPACED VALUES
       FOR LOG PSI.'
[25]  →(L1×⍳((-/¯2↑LP)=-/2↑LP)∧0>-/2↑LP←3↑,⎕),L2
[26] L3:'THE MATRIX OF EXPECTED FREQUENCIES IS NAMED  EF .',⎕←''
[27]  'LEAST MEMBER OF  EF  =  ';⌊/,EF←+/W
[28]  'STANDARDIZED RESIDUALS (SR):'
[29]  0.01×⌊0.5+100×SR←(EF*¯0.5)×M-EF
[30]  'CHI-SQUARED  =  ';+/,SR×SR;⎕←''
[31]  'DEGREES OF FREEDOM  =  ';¯1+×/ρLFR
    ∇
```

```
    ∇ MULTIPOLY M;I;K;N
[1]   →(ρI← 1 0)+(∧/,M≥0)∧(∧/,M=⌊M)∧(∧/2≤ρM)∧2≤K←ρρM
[2]   →0,ρ⎕←'NO GO.'
[3]   →3×⍳(2*K)>ρI←I,~I
[4]   →4×⍳(2×K)>ρρM←((⍳(ρρM)-K),((ρρM)-K)+(2+⍳K+1), 2 1)⍉M+.×(~N),[
      2.5] N←(⍳N)∘.>⍳¯1+N←¯1↑ρM
[5]   M←(N+1)÷2×((ρI),K↓ρM)ρM
[6]   K;'-VARIABLE INTERACTION--LOG CROSSPRODUCT RATIOS (LCR):'
[7]   0.01×⌊0.5+100×LCR←⍟(×/I/M)÷×/(~I)/M
[8]   'ESTIMATED STANDARD ERRORS (ESE):',⎕←''
[9]   0.01×⌊0.5+100×ESE←(2×+/÷M)*0.5
    ∇
```

HOWJACOBI

CHARACTERISTIC ROOTS AND VECTORS OF A SYMMETRIC MATRIX
Y←C JACOBI X

THE SECOND ARGUMENT (X) IS THE GIVEN MATRIX, TO BE TRANSFORMED TOWARDS DIAGONAL FORM BY JACOBI'S METHOD. THE FIRST ARGUMENT (C), A POSITIVE SCALAR, IS THE TOLERANCE FOR OFF DIAGONAL ELEMENTS IN THE TRANSFORMED MATRIX. THE EXPLICIT RESULT (Y) IS A THREE-DIMENSIONAL ARRAY WITH 2 PLANES. Y[1;;] IS THE TRANSFORMED MATRIX, AND SO 1 1 ⍉Y[1;;] IS THE VECTOR OF ROOTS, WHICH IS ALSO OUTPUT AS THE GLOBAL VARIABLE 'ROOTS'. THE CORRESPONDING VECTORS ARE THE COLUMNS OF Y[2;;] . THE GLOBAL SCALAR VARIABLE 'ITERATIONS' IS THE NUMBER OF PERFORMED STEPS OF THE JACOBI PROCESS.

USUALLY (IF THERE HAVE BEEN ENOUGH ITERATIONS) THE ROOTS APPEAR IN DECREASING ORDER.

TO SEE THE EFFECT OF REDUCING THE VALUE OF C , LET C1 AND C2 BE TOLERANCES, THE FIRST GREATER THAN THE SECOND. ENTER, FOR A GIVEN MATRIX X ,

```
Y1←C1 JACOBI X
ROOTS
ITERATIONS
Y2←C2 JACOBI Y1[1;;]
ROOTS
ITERATIONS
```

THE SECOND VERSION OF 'ROOTS' IS THE BETTER ONE. THE CORRESPONDING IMPROVED EIGENVECTORS ARE THE COLUMNS OF Y1[2;;]+.×Y2[2;;] .

5 DEC. 1972

```
    ∇ Y←C JACOBI X;A;I;J;K;M;N;R;S;Z
[1]    →2+(∧/C>0)∧(0=ρρC)∧((+/,C)>⌈/|(,X)-,⍉X)∧(4≤+/ρX)∧(=/ρX)∧2=ρρX
[2]    →0,ρ⎕←'NO GO.',Y←''
[3]    I←,(⍳N)∘.<⍳N←+/1↑ρY←X
[4]    R←(N,N)ρ1,NρITERATIONS←0
[5]   L1:→(C≥M←⌈/Z←|I/,Y)/L4
[6]    J←1+(M-K←1+N|¯1+M←(I\Z)⍳M)÷N
[7]    →(Y[J;J]≠Y[K;K])/L2
[8]    →L3,ρA←○¯0.25××Y[J;K]
[9]   L2:A←0.5×(○Y[J;J]<Y[K;K])+¯3○2×Y[J;K]÷Y[K;K]-Y[J;J]
[10]  L3:S←(N,N)ρ1,Nρ0
[11]   S[K;K,J]← 1 ¯1 ×S[J;J,K]← 2 1 ∘.○A
[12]   Y←(⍉R)+.×X+.×R←R+.×S
[13]   →L1,ρITERATIONS←ITERATIONS+1
[14]  L4:ROOTS← 1 1 ⍉Y
[15]   Y←Y,[0.5] R
    ∇
```

HOWPLOT

THREE FUNCTIONS FOR PLOTTING
(1) W←U SCATTERPLOT V
(2) DOWNPLOT V
(3) S ROWPLOT V

'SCATTERPLOT' IS USED FOR SCATTERPLOTTING CORRESPONDING MEMBERS OF TWO VECTORS. THE WHOLE CHARACTER ARRAY IS GENERATED IN THE WS AND MAY THEN BE PRINTED. 'DOWNPLOT' IS USED FOR PLOTTING MEMBERS OF ONE OR MORE VECTORS AGAINST THEIR INDEX NUMBERS. THE CHARACTER ARRAY IS GENERATED AND PRINTED LINE BY LINE; IT IS NOT STORED IN THE WS. 'ROWPLOT' GENERATES AND DISPLAYS IN ONE ROW THE CONTENTS OF EACH OF ONE OR MORE VECTORS. SEE ALSO 'HOWTRIPLEPLOT'.

(1) SCATTERPLOT. U AND V ARE VECTORS OF EQUAL LENGTH. CORRESPONDING MEMBERS U[J] AND V[J] ARE PLOTTED AS ABSCISSA AND ORDINATE OF A POINT. A SINGLE POINT IS SHOWN AS ○, TWO COINCIDENT POINTS AS ⊖, THREE OR MORE COINCIDENT POINTS AS ●. (YOU MAY CHANGE THIS CODE EASILY--SEE BELOW.)

TO GET A SINGLE PLOT DO NOT NAME AN EXPLICIT RESULT BUT CALL THUS:
U SCATTERPLOT V
TO GET A PLOT AND ALSO STORE THE SYMBOL ARRAY FOR MAKING A COPY LATER CALL THUS:
⎕←W←U SCATTERPLOT V

THE USER IS INTERROGATED ABOUT THE SIZE OF THE PLOT (NO. OF ABSCISSA VALUES, NO. OF ORDINATE VALUES), AND ABOUT WHETHER HE WANTS THE HORIZONTAL AND VERTICAL SCALES TO BE EQUAL. IF THE SCALES ARE NOT MADE EQUAL THEY ARE CHOSEN LIKE THIS. THE ABSCISSA VALUE SET IS EQUAL-SPACED, THE LEAST EQUAL TO (⌊/U), THE GREATEST TO (⌈/U), EXCEPT THAT IF THESE EXTREMES DO NOT HAVE THE SAME SIGN THE VALUE SET IS MINIMALLY TRANSLATED WITHOUT CHANGE OF SCALE SO THAT 0 BECOMES A MEMBER OF THE SET. SIMILARLY FOR ORDINATES. IF THE SCALES ARE MADE EQUAL, EITHER THE ABSCISSA VALUE SET OR THE ORDINATE VALUE SET MAY BE SHORTENED, TO AVOID UNNECESSARY BLANK COLUMNS OR ROWS IN THE PLOT. THE COORDINATES OF THE PLOTTED POINTS ARE ROUNDED TO THE NEAREST MEMBERS OF THE COORDINATE VALUE SETS. THE AXES ARE PRINTED AS ×'S, EXCEPT THAT IF 0 OCCURS IT IS SO SHOWN.

TO CHANGE THE PLOTTING CODE ALTER THE CHARACTER VECTOR IN LINE [1]. FOR EXAMPLE, TO SHOW MULTIPLICITIES BY NUMERALS, 1 TO 9, WITH X FOR 10 OR MORE, REPLACE '●⊖○' BY 'X987654321'. (MC SHOULD NOT CONTAIN A BLANK OR | OR -.)

THE TERMINAL IS SUPPOSED SET AT SINGLE SPACING, AND PAGE WIDTH AT THE MAXIMUM OF 130 CHARACTERS IF NEEDED. THE LAST LINE [29] SPECIFIES A HORIZONTAL SPACING OF THE DISPLAY; YOU MAY DELETE THIS LINE, IN WHICH CASE THE '64' IN LINE [7] SHOULD BE CHANGED TO '129'. THE LOCAL VARIABLE S IN LINE [12] SHOULD GIVE THE RELATIVE MAGNITUDES OF THE HORIZONTAL AND THE VERTICAL UNIT STEPS, AS PRINTED BY YOUR TERMINAL. WITH 6-PER-INCH LINEFEED, THE CORRECT SETTINGS FOR S ARE

FOR 12-PITCH CHARACTER SPACING, LAST LINE INTACT: S← 1 1
FOR 10-PITCH CHARACTER SPACING, LAST LINE INTACT: S← 6 5
FOR 12-PITCH CHARACTER SPACING, LAST LINE DELETED: S← 1 2
FOR 10-PITCH CHARACTER SPACING, LAST LINE DELETED: S← 3 5

(SAME FOR 'TSCP' , THE LAST LINE BEING NUMBERED [33].)

(2) DOWNPLOT. V IS EITHER (1) A VECTOR OR (2) A MATRIX. IF (1), V[J] IS PLOTTED AGAINST J. IF (2), THE ROWS ARE PLOTTED, I.E. V[;J] AGAINST J. POINTS ARE SHOWN AS ○; MULTIPLICITY IS NOT INDICATED. ← OR → MEANS AT LEAST ONE VALUE WAS OUTSIDE THE PLOTTING RANGE.

THE USER MUST SPECIFY SCALE AND LABELING (WITH THE GLOBAL VARIABLES DPS AND DPL), AND IS OFFERED INSTRUCTIONS.

TO CHANGE THE SYMBOLS (○, +, →) USED IN PLOTTING, ALTER LINES [33] *AND* [37]. *TO CHANGE THE TOP AND BOTTOM ROWS OF THE PLOT, MARKING THE ABSCISSA SCALE, ALTER THE DEFINITION OF AC IN LINE* [27]. *LABELING OF THE TOP ROW CAN BE INSERTED AS A NEW LINE* [26.1].

(3) *<u>ROWPLOT</u>. S IS A VECTOR SPECIFYING SCALE, JUST LIKE DPS; SEE INSTRUCTIONS IN 'DOWNPLOT'. V IS EITHER* (1) *A VECTOR OR* (2) *A MATRIX. IF* (1), *THE SET OF VALUES IS SHOWN AS POINTS ON A LINE. IF* (2), *EACH ROW IS SO SHOWN, ONE LINE PER ROW. MULTIPLICITY IS SHOWN AS IN 'SCATTERPLOT'. TO CHANGE THE PLOTTING CODE ALTER THE CHARACTER VECTOR IN LINE* [4].

25 *FEB*. 1972

```
    ∇ W←U SCATTERPLOT V;E;I;J;M;MC;N;R;S
[1]   →2+(1=ρρMC←'●⊖○')∧(1=ρρU)∧(1=ρρV)∧(N≥2)∧(+/ρU)=N←+/ρV
[2]   →ρ⎕←'ARGUMENTS SHOULD BE VECTORS OF EQUAL LENGTH NOT LESS THAN 2.',W←''
[3]   →4+∧/0<R←-/E← 2 2 ρ(⌈/U),(⌊/U),(⌈/V),⌊/V
[4]   →0,ρ⎕←'ONE ARGUMENT HAS ZERO RANGE.',W←''
[5]   'SIZE?  TYPE TWO NUMBERS, SUCH AS:  25 25'
[6]   →5×⍳(∨/M≠⌊M)∨(∨/1≥M)∨2≠ρM←,⎕
[7]   →8+64≥M[1]
[8]   →6,ρ⎕←'FIRST NUMBER TOO LARGE, TRY AGAIN.'
[9]   I←R÷M-J←1
[10]  'SAME SCALES?  Y OR N.'
[11]  →(('NY'=1↑,⍞)/ 13 12),10
[12]  E[;2]←E[;1]-I×¯1+M←1+⌈R÷I[1 2]←S×⌈/I÷S← 1 1
[13]  W←'×',((⌽M)ρ' '),[1] '×'
[14]  →L1×⍳0<×/ 1 2 ↑E←E÷I,[1.5] I
[15]  E[1;2]←(1-M[1])+E[1;1]←⌊0.5+E[1;1]
[16]  W[;(ρW)[2]-E[1;1]]←(M[2]ρ'|'),'0'
[17] L1:C,'EXTREME ABSCISSAS ARE:  ';⌽I[1]×E[1;]
[18]  'ABSCISSA UNIT STEP IS ';1↑I
[19]  →L2×⍳0<×/E[2;]
[20]  E[2;2]←(1-M[2])+E[2;1]←⌊0.5+E[2;1]
[21]  W[1+E[2;1];]←'0',M[1]ρ'-'
[22] L2:'EXTREME ORDINATES ARE:  ';⌽I[2]×E[2;]
[23]  'ORDINATE UNIT STEP IS ';I[2];C,C
[24]  U←(ρW)[2]-⌈E[1;1]-0.5+U÷I[1]
[25]  V←⌈0.5+E[2;1]-V÷I[2]
[26] L3:W[V[J];U[J]]←MC[1⌈¯1+MC⍳W[V[J];U[J]]]
[27]  →L3×⍳N≥J←J+1
[28]  U←V←''
[29]  W←((2×¯1↑ρW)ρ 0 1)\W
    ∇
```

```
    ∇ DOWNPLOT V;J;N;C;D;AC
[1]   →(((∧/1≤⍴V)∧ 1 2 =⍴⍴V)/ 3 4),2
[2]   →0,⍴⎕←'ARGUMENT WON''T DO.'
[3]   V←(1,⍴V)⍴V
[4]   'INSTRUCTIONS?  Y OR N'
[5]   →(('NY'=1↑,⍞)/ 21 6),4
[6]   'TWO GLOBAL VARIABLES MUST BE SPECIFIED,  DPS (DOWNPLOT SCALE)  AND DPL (D
      OWNPLOT'
[7]   'LABELING).  BOTH ARE VECTORS.  DPS INDICATES THE PLOTTING SCALE FOR ARGUM
      ENT VA-'
[8]   'LUES, LISTING THE LOWEST VALUE, THE STEP SIZE AND THE HIGHEST VALUE TO BE
       ACCOM-'
[9]   'MODATED.  HIGHEST MINUS LOWEST MUST BE DIVISIBLE BY STEP.   EVERY TENTH V
      ALUE IS'
[10]  'MARKED BY +, UNLESS LINE [27] IS ALTERED.  ENTER E.G. (45 PLOTTING VALUES
      ):'
[11]  '      DPS←8.1 0.1 12.5'
[12]  'DPL INDICATES THE LABELING OF THE INDEX NUMBERS, LISTING FOUR INTEGERS.
      THE 1ST'
[13]  'MUST BE POSITIVE AND NOT GREATER THAN THE 3RD.  THE DPL[1]''TH ROW OF THE
       PLOT IS'
[14]  'LABELED DPL[2],  AND THEREAFTER EVERY DPL[3]''TH ROW IS LABELED WITH AN I
      NCREMENT'
[15]  'OF DPL[4].  FOR SIMPLE COUNTING OF ROWS BY TENS ENTER:'
[16]  '      DPL←4⍴10'
[17]  'WITH DPS AND DPL SPECIFIED, RESUME EXECUTION OF DOWNPLOT BY TYPING:'
[18]  '      →20'
[19]  SΔDOWNPLOT←¯9+⍴⎕←'SEE HOWPLOT FOR FURTHER INFO.'
[20]  SΔDOWNPLOT←0
[21]  →23-J←∨/(3=⍴DPS)∧(∧/0≠¯1↓1↓,DPS)∧1=⍴⍴DPS
[22]  →23+(N=⌊N)∧(125≥N)∧2≤N←1+(-/DPS[3 1])÷DPS[2]
[23]  →0,⍴⎕←'DPS WON''T DO.'
[24]  →26-(∧/,DPL=⌊DPL)∧(4=+/⍴DPL)∧1=⍴⍴DPL
[25]  →26+(1≤DPL[1])∧≥/DPL[3 1],⎕←''
[26]  →0,⍴⎕←'DPL WON''T DO.'
[27]  '    +',(N←⌊N)⍴AC←'---------+'
[28] L1:C←(N+2)⍴1
[29]  C[1⌈(⌊2.5+(V[;J]-DPS[1])÷DPS[2])⌊N+2]←2
[30]  C[2]←3⌊C[2]+2×C[1]-1
[31]  C[N+1]←4⌊C[N+1]+3×C[N+2]-1
[32]  →L3×⍳0=DPL[3]|J-DPL[1]
[33]  '    |',' ○↔'[1↓¯1↓C]
[34] L2:→L1×⍳(⍴V)[2]≥J←J+1
[35]  →0,⍴⎕←'    +',N⍴AC
[36] L3:D←(4⍴10)⊤DPL[2]+DPL[4]×(J-DPL[1])÷DPL[3]
[37]  →L2,⍴⎕←'0123456789 '[1+D+(10×(D=0)∧.∨(⍳4)∘.>⍳3),0],'|',' ○↔'[1↓¯1↓C]
    ∇
```

HOWINTEGRATE

ONE-DIMENSIONAL DEFINITE INTEGRALS
Z←H INTEGRATE A

THE FIRST ARGUMENT (H) IS A STEP SIZE, SCALAR. THE SECOND ARGUMENT (A) IS A VECTOR OF 2 OR MORE LIMITS OF INTEGRATION, IN ASCENDING ORDER, WITH DIFFERENCES ALL DIVISIBLE BY H . THE EXPLICIT RESULT (Z) IS A VECTOR OF LENGTH 1 LESS THAN THE LENGTH OF A , LISTING THE DEFINITE INTEGRALS FROM A[1] *TO EACH OF THE OTHER MEMBERS OF A . THE FUNCTION TO BE INTEGRATED IS ASKED FOR, AND MUST BE EXPRESSED IN TERMS OF AN ARGUMENT X (LOCAL VARIABLE). TWO-POINT GAUSSIAN QUADRATURE IS USED IN EACH INTERVAL OF WIDTH H .*

FOR EXAMPLE, TO INTEGRATE SIN X FROM 0 *TO EACH OF* 1 2 3 *AND COMPARE THE RESULT WITH* 1 - *COS X :*

```
      0.1 INTEGRATE ¯1+⍳4
FUNCTION OF X TO BE INTEGRATED?
⎕:
      1○X
0.4596976835  1.416146804  1.989992451
      1-2○⍳3
0.4596976941  1.416146837  1.989992497
```

29 *NOV.* 1972

```
     ∇ Z←H INTEGRATE A;N;X
[1]    →3-(∧/2≤ρA)∧(1=ρρA)∧(∧/,H>0)∧0=ρρH
[2]    →3+(∧/0<N-0,¯1↓N)∧∧/N=⌊N←((1↓A)-1↑A)÷H
[3]    →0,ρ⎕←'NO GO.',Z←''
[4]    'FUNCTION OF X TO BE INTEGRATED?'
[5]    X←(A[1]+H×⍳¯1↑N)∘.-(H÷2)×1+ 1 ¯1 ÷3*0.5
[6]    Z←(H÷2)×(N∘.≥⍳¯1↑N)+.×+/⎕
     ∇
```

```
     ∇ S ROWPLOT V;J;N;C;MC
[1]    →(((∧/1≤ρV)∧ 1 2 =ρρV)/ 3 4),2,⍞←⍳0
[2]    →ρ⍞←'SECOND ARGUMENT WON''T DO.'
[3]    V←(1,ρV)ρV
[4]    →6-J←∨/(3=ρS)∧(1=ρρS)∧1=ρρMC←⌽'⍟⊖○'
[5]    →6+(125≥N)∧(N=⌊N)∧2≤N←1+(-/S[3 1])÷S[2]
[6]    →ρ⍞←'FIRST ARGUMENT WON''T DO.'
[7]    C←(ρMC)⌊+/(⍳N+2)∘.=1⌈(⌊2.5+(V[J;]-S[1])÷S[2])⌊N+2
[8]    C[2]←(0 1 =1⌊1↑C)/C[2],¯2
[9]    C[N+1]←(0 1 =1⌊¯1↑C)/C[N+1],¯1
[10]   5ρ' ';('↔ ',MC)[3+1↓¯1↓C]
[11]   →7×⍳(1↑ρV)≥J←J+1
[12]   5ρ' ';Nρ'---------+'
     ∇
```

HOWTRIPLEPLOT

FUNCTIONS FOR PLOTTING
(1) *Z TDP V*
(2) *W←U TSCP V*
(3) *ROWCOLDISPLAY I*

IN THESE VERSIONS OF FUNCTIONS DESCRIBED IN 'HOWPLOT', A THIRD DIMENSION IS SUGGESTED BY VARYING THE SYMBOL USED IN PLOTTING THE POINTS.

(1) *TDP (TRIPLE-DOWNPLOT) IS A MODIFICATION OF 'DOWNPLOT'. THE GLOBAL VARIABLES DPS AND DPL MUST BE SPECIFIED BEFORE EXECUTION. THE FIRST ARGUMENT (Z) IS A CHARACTER SCALAR OR VECTOR SHOWING WHAT SYMBOLS ARE TO BE PLOTTED. IF Z IS SCALAR, THE SAME SYMBOL WILL BE PLOTTED EVERY TIME. THE CALL*

```
'○' TDP V
```

YIELDS THE SAME RESULT AS THE CALL

```
DOWNPLOT V
```

WITH THE EXCEPTIONS THAT INSTRUCTIONS ARE NOT OFFERED AND THAT IF ANY MEMBERS OF V FALL OUTSIDE THE RANGE SPECIFIED BY DPS THEY ARE MISSED (NO WARNING ← OR →).

IF Z IS NOT SCALAR, IT MUST BE A CHARACTER VECTOR OF LENGTH EQUAL TO ¯1↑⍴V *. Z[J] IS THE SYMBOL USED IN PLOTTING V[J] (IF V IS A VECTOR) OR V[;J] (IF V IS A MATRIX) AGAINST J.*

TO CHANGE THE TOP AND BOTTOM ROWS OF THE PLOT MARKING THE ABSCISSA SCALE, ALTER THE DEFINITION OF AC IN LINE [12]. *LABELING OF THE TOP ROW CAN BE INSERTED AS A NEW LINE* [11.1].

(2) *TSCP (TRIPLE-SCATTERPLOT) IS A MODIFICATION OF 'SCATTERPLOT'. THE ARGUMENTS (U, V) SPECIFY ABSCISSAS AND ORDINATES, AS BEFORE. ALL COINCIDENT POINTS ARE SHOWN AS* ⋆ . *AFTER THE INTERROGATIONS ABOUT FINENESS OF PLOTTING AND EQUAL SCALES, THE USER IS ASKED FOR THE SYMBOLS TO BE USED. IF HE REPLIES WITH ONE SYMBOL IN QUOTES, SUCH AS* '○'*, THE RESULT WILL BE LIKE THAT OF 'SCATTERPLOT' EXCEPT THAT ALL MULTIPLE POINTS ARE SHOWN BY THE ONE SYMBOL* ⋆ . *OTHERWISE HE MUST REPLY WITH A CHARACTER VECTOR OF LENGTH EQUAL TO* ⍴U *AND* ⍴V *, NONE OF THE CHARACTERS BEING A BLANK OR* ⋆ . *FOR A VECTOR V , THE CALL*

```
(⍳⍴V) TSCP V
```

WITH Z AS THE ANSWER TO THE THIRD QUESTION YIELDS A PLOT EQUIVALENT TO

```
Z TDP V
```

IF THE SCALES ARE SUITABLY MATCHED. BECAUSE THE MINUS SIGN MAY BE NEEDED AS A PLOTTING SYMBOL, ZERO LINES ARE NOT INDICATED IN THE PLOT WITH | *OR* - *, AS THEY ARE IN 'SCATTERPLOT'.*

(3) *ROWCOLDISPLAY IS A SPECIAL FUNCTION, SUBSTITUTING FOR 'TSCP', THAT MAY BE USED TO DISPLAY THE OUTPUT OF 'ROWCOL'. ABSCISSAS AND ORDINATES ARE THE COLUMN EFFECTS AND THE ROW EFFECTS. THE SYMBOLS PLOTTED REPRESENT THE RESIDUALS GRADED ON A 7-POINT SCALE FROM LARGE-NEGATIVE TO LARGE-POSITIVE:* M̲, M, -, ∘, +, P, P̲.

AT THE EXPENSE OF ACCURACY IN POSITIONING THE PLOTTED POINTS, COINCIDENCES OF ROWS OR COLUMNS ARE AVOIDED THROUGH A UNIT MINIMUM SEPARATION, EVEN WHEN TWO ROW EFFECTS OR COLUMN EFFECTS ARE EQUAL.

THE ARGUMENT (I) IS THE CHANGE IN COLUMN EFFECT REPRESENTED BY A UNIT HORIZONTAL DISPLACEMENT. TRY EXECUTING THE FUNCTION WITH A LARGISH VALUE FOR I, NOT LESS THAN ((⌈/CE)-⌊/CE)÷20 . *IF THE DISPLAY IS TOO SMALL REPEAT WITH A SMALLER VALUE FOR I .*

THE FUNCTION REFERS TO THE GLOBAL VARIABLES CE, RE, RY GENERATED BY 'ROWCOL', BUT THESE SHOULD FIRST BE REARRANGED BY CALLING THE FUNCTION 'ROWCOLPERMUTE' (NO ARGUMENTS OR EXPLICIT RESULT), WHICH PERMUTES THE ROWS AND COLUMNS SO THAT CE IS IN ASCENDING ORDER AND RE IN DESCENDING ORDER.

ADJUSTMENTS IN LINE [1]: *TO SUPPRESS HORIZONTAL DOUBLE SPACING SET* (*D*←1). *FOR* 6-*PER-INCH LINEFEED*, 12-*PITCH CHARACTER SPACING, IF* *D*←2 *THEN* *S*← 1 1 , *BUT IF* *D*←1 *THEN* *S*← 1 2 . *FOR* 10-*PITCH SPACING, SET* *S*← 6 5 *AND* *S*← 3 5 , *RESP.*

2F *FEB.* 1972

```
    ∇ W←U TSCP V;E;I;J;M;N;R;S;Z̲
[1]   →2+(1=ρρU)∧(1=ρρV)∧(N≥2)∧(+/ρU)=N←+/ρV
[2]   →ρ⎕←'ARGUMENTS SHOULD BE VECTORS OF EQUAL LENGTH NOT LESS THAN 2.',W←''
[3]   →4+∧/0<R←-/E← 2 2 ρ(⌈/U),(⌊/U),(⌈/V),⌊/V
[4]   →0,ρ⎕←'ONE ARGUMENT HAS ZERO RANGE.',W←''
[5]   'SIZE?  TYPE TWO NUMBERS, SUCH AS:  25 25'
[6]   →5×⍳(∨/M≠⌊M)∨(∨/1≥M)∨2≠ρM←,⎕
[7]   →8+64≥M[1]
[8]   →6,ρ⎕←'FIRST NUMBER TOO LARGE, TRY AGAIN.'
[9]   I←R÷M-J←1
[10]  'SAME SCALES?  Y OR N.'
[11]  →(('NY'=1↑,⍞)/ 13 12),10
[12]  E[;2]←E[;1]-I×¯1+M←1+⌈R÷I[1 2]←S×⌈/I÷S← 1 1
[13]  W←'×',((⌽M)ρ' '),[1] '×'
[14]  →L̲1×⍳0<×/ 1 2 ↑E←E÷I,[1.5] I
[15]  E[1;2]←(1-M[1])+E[1;1]←⌊0.5+E[1;1]
[16]  W[1↑ρW;(ρW)[2]-E[1;1]]←'0'
[17] L̲1:'SYMBOLS TO BE USED?'
[18]  →L̲2×⍳0<×/E[2;]
[19]  E[2;2]←(1-M[2])+E[2;1]←⌊0.5+E[2;1]
[20]  W[1+E[2;1];1]←'0'
[21] L̲2:→(((~∨/,Z̲∊' *')∧(∧/N=ρZ̲)∧ 0 1 =ρρZ̲←⎕)/L̲3+ 1 2),L̲3
[22] L̲3:→(L̲3-1),ρ⎕←'NO GOOD, TRY AGAIN.'
[23]  Z̲←NρZ̲
[24]  C̲,'EXTREME ABSCISSAS ARE:  ';⌽I[1]×E[1;]
[25]  'ABSCISSA UNIT STEP IS ';1↑I
[26]  'EXTREME ORDINATES ARE:  ';⌽I[2]×E[2;]
[27]  'ORDINATE UNIT STEP IS ';I[2];C̲,C̲
[28]  U←(ρW)[2]-⌈E[1;1]-0.5+U÷I[1]
[29]  V←⌈0.5+E[2;1]-V÷I[2]
[30] L̲4:W[V[J];U[J]]←1↑((W[V[J];U[J]]≠' ')/'*'),Z̲[J]
[31]  →L̲4×⍳N≥J←J+1
[32]  U←V←Z̲←''
[33]  W←((2×¯1↑ρW)ρ 0 1)\W
    ∇
```

HOWMAX

MAXIMIZATION OF A FUNCTION OF ONE VARIABLE
Z←X MAX Y

THE ARGUMENTS ARE VECTORS OF LENGTH 3, *THE FIRST HAVING NO TWO MEMBERS EQUAL. THE EXPLICIT RESULT IS THE COORDINATES OF THE VERTEX OF A PARABOLA WITH VERTICAL AXIS, THAT GOES THROUGH THE* 3 *POINTS WHOSE ABSCISSAS ARE THE FIRST ARGUMENT AND ORDINATES THE SECOND ARGUMENT. IF THE PARABOLA IS CONVEX FROM BELOW, THE VERTEX IS THE MINIMUM INSTEAD OF THE MAXIMUM POINT, AND THE WORD 'MINIMUM.' APPEARS IN THE OUTPUT.*

29 *NOV*. 1972

```
    ∇ Z←X MAX Y;A;B;M
[1]   →3-(3=+/ρX)∧(3=+/ρY)∧(1=ρρX)∧1=ρρY
[2]   →3+(∧/(⍒X)=⌽⍋X)∧=/Z← 0 0
[3]   →ρ⎕←'ARGUMENTS WON''T DO.',Z←''
[4]   A←Y⌹1,X,[1.5] X×X←X-M←(+/X)÷3
[5]   →((A[3]=0),A[3]<0)/ 9 7
[6]   'MINIMUM.'
[7]   Z[1]←M+B←(÷/1↓A)÷¯2
[8]   →0,Z[2]←A[1 2]+.×1,B÷2
[9]   'NO TURNING POINT.',Z←''
    ∇
```

```
    ∇ Z TDP V;J;N;AC;C;D
[1]   →(((∧/1≤ρV)∧ 1 2 =ρρV)/ 3 4),2
[2]   →0,ρ⍞←'ARGUMENTS WON''T DO.'
[3]   V←(1,ρV)ρV
[4]   →((0 1 =ρρZ)/5,2+4×∨/(ρZ)=¯1↑ρV),2
[5]   Z←(ρV)[2]ρZ
[6]   →8-J←∨/(3=ρDPS)∧(∧/0≠¯1↓1↓,DPS)∧1=ρρDPS
[7]   →8+(N=⌊N)∧(125≥N)∧2≤N←1+(-/DPS[3 1])÷DPS[2]
[8]   →0,ρ⍞←'DPS WON''T DO.'
[9]   →11-(∧/,DPL=⌊DPL)∧(4=+/ρDPL)∧1=ρρDPL
[10]  →11+(1≤DPL[1])∧≥/DPL[3 1],⍞←''
[11]  →0,ρ⍞←'DPL WON''T DO.'
[12]  '     +',(N←⌊N)ρAC←'---------+'
[13] L1:C←(N+2)ρ' '
[14]  C[1⌈(⌊2.5+(V[;J]-DPS[1])÷DPS[2])⌊N+2]←Z[J]
[15]  →L3×⍳0=DPL[3]|J-DPL[1]
[16]  '     |',1↓¯1↓C
[17] L2:→L1×⍳(ρV)[2]≥J←J+1
[18]  →0,ρ⍞←'     +',NρAC
[19] L3:D←(4ρ10)⊤DPL[2]+DPL[4]×(J-DPL[1])÷DPL[3]
[20]  →L2,ρ⍞←'0123456789 '[1+D+(10×(D=0)∧.∨(⍳4)∘.>⍳3),0],'|',1↓¯1↓C
    ∇
```